Robust Meta-Representation Learning via Global Label Inference and Classification

Ruohan Wang, John Isak Texas Falk, Massimiliano Pontil, and Carlo Ciliberto

***Abstract*—Few-shot learning (FSL) is a central problem in meta-learning, where learners must efficiently learn from few labeled examples. Within FSL, feature pre-training has become a popular strategy to significantly improve generalization performance. However, the contribution of pre-training to generalization performance is often overlooked and understudied, with limited theoretical understanding. Further, pre-training requires a consistent set of global labels shared across training tasks, which may be unavailable in practice. In this work, we address the above issues by first showing the connection between pre-training and meta-learning. We discuss why pre-training yields more robust meta-representation and connect the theoretical analysis to existing works and empirical results. Second, we introduce Meta Label Learning (MeLa), a novel meta-learning algorithm that learns task relations by inferring global labels across tasks. This allows us to exploit pre-training for FSL even when global labels are unavailable or ill-defined. Lastly, we introduce an augmented pre-training procedure that further improves the learned meta-representation. Empirically, MeLa outperforms existing methods across a diverse range of benchmarks, in particular under a more challenging setting where the number of training tasks is limited and labels are task-specific.**

***Index Terms*—Few-Shot image classification, learning with limited labels, meta-learning, representation learning.**

I. Introduction

DEEP neural networks have facilitated transformative advances in machine learning in various areas [e.g., [7], [23], [26], [34], [47], [67]. However, state-of-the-art models typically require labeled datasets of extremely large scale, which are prohibitively expensive to curate. When training data is scarce, neural networks often overfits with degraded performance. Few-shot learning (FSL) aims to address this loss in performance by developing algorithms and architectures capable of learning from few labeled samples.

Meta-learning [27], [74] is a popular class of algorithms to tackle FSL. Broadly, meta-learning seeks to learn transferable knowledge over many FSL tasks, and to apply such knowledge to novel ones. For instance, Model Agnostic Meta Learning (MAML) [17] learns a prior over the model initialization that is suitable for fast adaptation. Existing meta-learning methods for tackling FSL may be loosely classified into three categories; optimization [e.g.[6], [17], [80], metric learning [e.g., [68], [70], [75], and model-based methods [e.g.[24], [53], [62]. The diversity of existing strategies poses a natural question: can we derive any "meta-insights" from them to facilitate the design of future methods?

Among the existing methods, several trends have emerged for designing robust few-shot meta-learners. Chen et al. [8] observed that data augmentation and deeper networks significantly improves generalization performance. The observations have since been widely adopted [e.g.[4], [36], [71]. *Network pre-training* has also become ubiquitous [e.g.[15], [60], [77], [80], and dominates state-of-the-art models. Sidestepping the task structure and episodic training of meta-learning, pre-training learns (initial) model parameters by merging all FSL tasks into one "flat" dataset of labeled samples followed by standard multi-class classification. The model parameters may be further fine-tuned to improve performance.

Despite its popularity, the limited theoretical understanding of pre-training leads to diverging interpretations of existing methods. Many meta-learning methods consider pre-training as nothing but a standard pre-processing step, and attribute the observed performance almost exclusively to their respective algorithmic and network design [e.g.[62], [82], [84]. However, extensive empirical evidence suggests that pre-training is crucial for model performance [78], [80]. Tian et al. [71] demonstrated that simply learning task-specific linear classifiers over the pre-trained representation outperforms various meta-learning strategies. Wertheimer et al. [80] further showed that earlier FSL methods also benefit from pre-training, resulting in improved performance.

In this work we contributes a unified perspective by showing that pre-training directly relates to meta-learning by minimizing an upper bound on the meta-learning loss. In particular, we show

Manuscript received 9 December 2022; revised 2 October 2023; accepted 18 October 2023. Date of publication 27 October 2023; date of current version 6 March 2024. The work of Ruohan Wang was supported by the funding from Career Development Fund under Grant C210812045 from A*STAR Singapore. The work of John Isak Texas Falk was supported by the funding from the computer science department with UCL. This work of Massimiliano Pontil was supported in part from the PNRR MUR Project PE000013 CUP J53C22003010006 "Future Artificial Intelligence Research (FAIR)", funded by the European Union - NextGenerationEU. The work of Carlo Ciliberto was supported in part by from the Royal Society under Grant SPREM RGS-R1-201149, and in part by Amazon Research Award (ARA). Recommended for acceptance by C. G. M. Snoek. *(Corresponding author: Massimiliano Pontil.)*

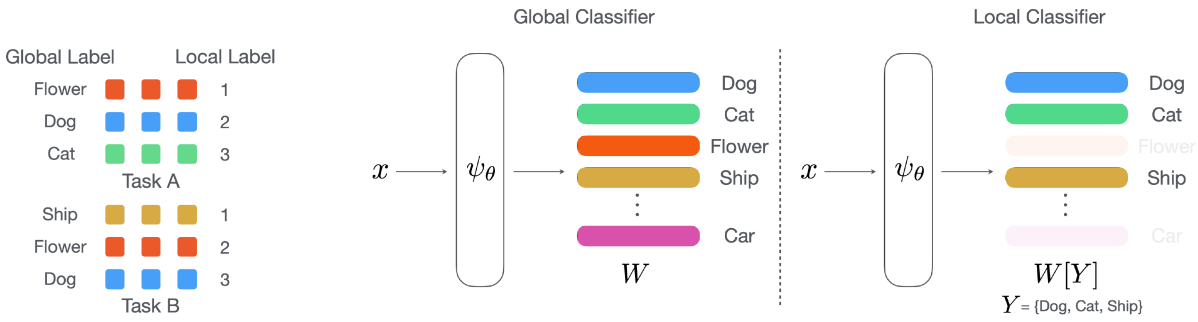
Ruohan Wang is with the Institute for Infocomm Research (I^{2}R), Agency for Science, Technology and Research (A*STAR), Singapore 138632 (e-mail: wang_ruohan@i2r.a-star.edu.sg).

John Isak Texas Falk and Massimiliano Pontil are with the Centre for Artificial Intelligence, Computer Science Department, University College London, WC1E 6BT London, U.K., and also with the CSML, Isitituto Italiano di Tecnologia, 16121-16167 Genova, Italy (e-mail: ucabitf@ucl.ac.uk; massimiliano.pontil@iit.it).

Carlo Ciliberto is with the Centre for Artificial Intelligence, Computer Science Department, University College London, WC1E 6BT London, U.K. (e-mail: cciliber@gmail.com).

This article has supplementary downloadable material available at https://doi.org/10.1109/TPAMI.2023.3328184, provided by the authors.

Digital Object Identifier 10.1109/TPAMI.2023.3328184



(a) **Global vs. Local labels** (b) **Global to Local Classifier**

Fig. 1. (a) Colored squares represent samples. Tasks A and B can be "merged" meaningfully using global labels, but not local ones. (b) A global classifier can be used as local classifiers given the indices Y of the intended classes to predict.

that pre-training achieves a smaller expected error and enjoys a better convergence rate compared to its meta-learning counterpart. More broadly, we connect pre-training to conditional meta-learning [11], [77], which has favorable theoretical properties including tighter bounds. Our result provide a principled justification of why pre-training yields a robust meta-representation for FSL, and the associated performance improvement.

Motivated by this result, we propose an augmentation procedure for pre-training that quadruples the number of training classes by considering rotations as novel classes and classifying them jointly. This significantly increases the size of training data and leads to robust representations. We empirically demonstrate that the augmentation procedure consistently performs better across different benchmarks.

The standard FSL setting [e.g.[6], [17], [48] assumes access to a collection of tasks (i.e., the meta-training set) for training data. To perform pre-training, meta-training tasks must be merged into a flat dataset (see Section II-C for a formal definition), which implicitly assumes access to some notion of *global labels* shared across all tasks. However, global labels may be unavailable, such as when each task is independently labeled with only *local labels*. This renders naive task merging and pre-training infeasible (see Fig. 1(a)). Independent task annotation is a more realistic and general assumption, capturing scenarios when training tasks are collected *organically* from different sources rather than generated *synthetically* from a base dataset. Practical scenarios where naive task merging is infeasible include non-descriptive task labels (e.g., numerical ones) or concept overlap (e.g., marine animals vs mammals) among labels.

To tackle independent task annotation, we propose **Me**ta **La**bel Learning (MeLa), a novel algorithm that automatically infers a notion of *latent* global labels consistent with local task constraints. The inferred labels enable us to exploit pre-training for FSL, and to bridge the gap between experimental settings with or without access to global labels. Empirically, we demonstrate that MeLa is competitive with pre-training on oracle labels.

For experiments, we introduce a new Generalized FSL (GFSL) setting. In addition to independent task annotation, we also adopt a fixed-size meta-training set and enforce no repetition of samples across tasks. This challenging setting evaluates how efficiently meta-learning algorithms generalize from limited number of tasks, and prevents the algorithms from trivially uncover task relations by implicitly matching identical samples across tasks. We empirically show that MeLa performs robustly in both standard and GFSL settings, and clearly outperforms state-of-the-art models in the latter.

We summarize the main contributions below:

- We prove that pre-training relates to meta-learning as a loss upper bound. Consequently, minimizing the pre-training loss is a viable proxy for tackling meta-learning problems. Additionally, we identify meta-learning regimes where pre-training offers a clear improvement with respect to sample complexity. This theoretical analysis provides a principled explanation for pre-training's empirical advantage.
- We propose MeLa, a general algorithm for inferring latent global labels from meta-training tasks. It allows us to exploit pre-training when global labels are absent or ill-defined.
- We propose an augmented pre-training procedure for FSL and a GFSL experimental setting.
- Extensive experiments demonstrate the robustness of MeLa. Detailed ablations provide deeper understanding of the model.

Extension of [78]: This paper is an extended version of [78] with the following contributions in addition to those of the original work: i) a deeper theoretical insight into the role of pre-training from the perspective of the risk (rather than the empirical risk as in [78]), and quantifying its benefit in terms of sample complexity, ii) the augmented training procedure for FSL, iii) the GFSL experimental setting, iv) significantly more empirical evidence to support the proposed algorithm.

II. Background

We formalize FSL as a meta-learning problem and review related methods. We also discuss the pre-training procedure adopted by many FSL methods.

A. Few-Shot Learning Using Meta-Learning

FSL [16] considers a meta-training set of tasks $\mathcal{T} = \{(S_t, Q_t)\}_{t=1}^{T}$, with *support set* $S_t = \{(x_j, y_j)\}_{j=1}^{n_s}$ and *query set* $Q_t = \{(x_j, y_j)\}_{j=1}^{n_q}$ sampled from the same distribution. Typically, S_t and Q_t contain a small number of samples n_s

and n_q respectively. We denote by $\mathcal{D}$ the space of datasets of the form S_t or Q_t.

The meta-learning formulation for FSL aims to find the best *base learner* $\text{Alg}(\theta, \cdot) : \mathcal{D} \to \mathcal{F}$ that takes as input support sets S, and outputs predictors $f = \text{Alg}(\theta, S)$, such that predictions $y = f(x)$ generalize well on the corresponding query sets Q. The base learner is meta-parameterized by $\theta \in \Theta$. Formally, the meta-learning objective for FSL is

$$\mathbb{E}_{(S,Q)\in\mathcal{T}} \mathcal{L}\left(\text{Alg}(\theta, S),\, Q\right), \tag{1}$$

where $\mathbb{E}_{(S,Q)\in\mathcal{T}} \triangleq \frac{1}{|\mathcal{T}|}\sum_{(S,Q)\in\mathcal{T}}$ is the empirical average over the meta-training set $\mathcal{T}$. The *task loss* $\mathcal{L} : \mathcal{F} \times \mathcal{D} \to \mathbb{R}$ is the empirical risk of the learner f over query sets, based on some *inner loss* $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$, where $\mathcal{Y}$ is the space of labels,

$$\mathcal{L}(f, D) \;=\; \mathbb{E}_{(x,y)\in D}\left[\ell(f(x), y)\right]. \tag{2}$$

Equation (1) is sufficiently general to describe most existing methods. For instance, model-agnostic meta-learning (MAML) [17] parameterizes a model $f_\theta : \mathcal{X} \to \mathcal{Y}$ as a neural network, and $\text{Alg}(\theta, D)$ performs one (or more) steps of gradient descent minimizing the empirical risk of f_θ on D. Formally, given a step-size $\eta > 0$,

$$f_{\theta'} = \text{Alg}(\theta, D) \quad \text{with} \quad \theta' = \theta - \eta\,\nabla_\theta \mathcal{L}(f_\theta, D). \tag{3}$$

Clearly, base learners $\text{Alg}(\theta, \cdot)$ is key to model performance and various strategies have been explored. Our proposed method is most closely related to meta-representation learning [6], [18], [36], [55], which parameterizes the base learner as $A(\theta, D) = w(g_\theta(D))g_\theta(\cdot)$, consisting of a global feature extractor $g_\theta : \mathcal{X} \to \mathbb{R}^m$ and a task-adaptive classifier $w : \mathcal{D} \to \{f : \mathbb{R}^m \to \mathcal{Y}\}$ that optimizes the following

$$\min_{\theta\in\Theta}\; \mathbb{E}_{(S,Q)\in\mathcal{T}}\left[\mathcal{L}\left(w(g_\theta(S)), g_\theta(Q)\right)\right] \tag{4}$$

where $g_\theta(D) \triangleq \{(g_\theta(x), y)\,|\,(x, y) \in D\}$ is the embedded dataset. Equation (4) specializes (1) by learning a feature extractor g_θ shared (and fixed) among tasks. Only the classifier returned by $w(\cdot)$ adapts to the current task, in contrast to having the entire model $f_\theta : \mathcal{X} \to \mathcal{Y}$ adapted (e.g., (3) for MAML). While this may appear to restrict model adaptability, [55] has demonstrated that meta-representation learning matches MAML's performance. Moreover, they showed that feature reuse is the dominant contributor to the generalization performance rather than adapting the representation to the task at hand.

The task-adaptive classifier $w(\cdot)$ may take various forms, including nearest neighbor [68], ridge regression classifier [6], embedding adaptation with transformer models [84], and Wasserstein distance metric [85]. In particular, the ridge regression estimator

$$w_{\text{ridge}}(D) = \text{argmin}_W\, \mathbb{E}_{(x,y)\in D}\, \|Wx - y\|^2 + \lambda_1 \|W\|_F^2, \tag{5}$$

where $\|\cdot\|_F$ is the Frobenius norm, admits a differentiable closed-form solution and is computationally efficient for optimizing (4).

B. Conditional Meta-Learning

Conditional formulations of meta-learning [11], [77] extends (1) by considering base learners of the form $\text{Alg}(\tau(Z), S)$, where the meta-parameters $\theta = \tau(Z)$ is conditioned on some "contextual" information $Z \in \mathcal{Z}$ about the task S. Assuming each task in the meta-training set $\mathcal{T}$ to be equipped with such contextual information, (1) can be re-expressed as

$$\min_{\tau:\mathcal{Z}\to\Theta}\; \mathbb{E}_{(S,Q,Z)\in\mathcal{T}}\, \mathcal{L}\left(\text{Alg}(\tau(Z), S),\, Q\right), \tag{6}$$

namely the problem of learning a function $\tau : \mathcal{Z} \to \Theta$, which maps contextual information $Z \in \mathcal{Z}$ (e.g., a textual meta-description of the task/dataset) to a good task-specific base learner with parameters $\theta = \tau(Z)$.

The conditional formulation seeks to capture complex (e.g., multi-modal) distributions of meta-training tasks, and uses a unique base learner tailored to each one. In particular, [62], [76], [83] directly learn data-driven mappings from target tasks to meta-parameters, and [31] conditionally transforms feature representations based on a metric space trained to capture inter-class dependencies. Alternatively, [30] considers a mixture of hierarchical Bayesian models over the parameters of meta-learning models in order to condition on target tasks. In [77], Wang et al. showed that conditional meta-learning can be interpreted as a structured prediction problem and proposed a method leveraging structured prediction. From a more theoretical perspective, Denevi et al. [11], [12] proved that conditional meta-learning is theoretically advantageous compared to unconditional approaches by incurring smaller excess risk and being less prone to negative transfer. As we will discuss in Section III, conditional meta-learning is closely related to our theoretical analysis on feature pre-training.

C. Feature Pre-Training

Feature pre-training has been widely adopted in the recent meta-learning literature [e.g.[1], [4], [8], [45], [50], [58], [69], [77], [80], [82], [84], [85], and is arguably one of the key contributors to performance of state-of-the-art models. Instead of directly learning the feature extractor g_θ by optimizing (4), pre-training first learns a feature extractor via standard supervised learning.

Formally, the meta-training set $\mathcal{T}$ is "flattened" into D_{global} by merging all tasks:

$$D_{\text{global}} = D(\mathcal{T}) = \{(x_i, y_i)\}_{i=1}^N = \bigcup_{(S,Q)\in\mathcal{T}} (S \cup Q), \tag{7}$$

where we have re-indexed the (x_i, y_i) samples from $i = 1$ to N (the cumulative number of points from all support and query sets) to keep the notation uncluttered. Pre-training then learns the embedding function g_θ on D_{global} using the standard cross-entropy loss ℓ_{ce} for multi-class classification:

$$(W_N^{\text{pre}}, \theta_N^{\text{pre}}) = \underset{\theta,W}{\text{argmin}}\; \mathcal{L}(Wg_\theta, D_{\text{global}}). \tag{8}$$

where W is the linear classifier over all classes. After pre-training, the feature extractor is either fixed [e.g.[62], [71],

[77], [84], [85] or further adapted [e.g., [4], [5], [21], [58] via meta-learning.

There is limited theoretical understanding and consensus on the effect of pre-training in FSL. In [4], [62], [82], the pre-training is only considered a standard pre-processing step for encoding the raw input and model performance is predominantly attributed to the proposed meta-learning algorithms. In [21] the authors similarly argued that meta-trained features are better than pre-trained ones, observing that adapting the pre-trained features with several base learners resulted in worse performance compared to the meta-learned features. In contrast, however, several works also empirically demonstrated that pre-training contributes significantly towards performance. [71] showed that combining the pre-trained features with suitable base learners already outperforms various meta-learning methods, while [15] observed that pre-training dominates top entries in 2021 Meta-learning Challenge.

We conclude by noting that recently pre-training has also been successfully applied to large-scale multi-modal settings combining visual and language input, enabling zero-shot learning [54], more flexible few-shot learning [2] (i.e., tasks may be described using free text), and generating more samples to augment FSL [81]. While this line of work further showcases the efficacy of pre-training strategies for FSL, in this work we focus on few shot learning settings with a single input modality.

III. Pre-Training as Meta-Learning

In this section, we characterize how feature pre-training relates to meta-learning as a loss upper bound. More precisely, we show that pre-training induces a special base learner with its corresponding meta-learning loss upper bounded by the cross-entropy loss ℓ_{ce}. Consequently, pre-training already produces a meta-representation suitable for FSL, matching the empirical results from [71], [84]. In addition, we show that pre-training incurs a smaller risk compared to its meta-learning counterpart, and more generally induces a conditional formulation that exploits contextual information for more robust learning.

A. Notation and Problem Setting

We consider a few-shot classification setting with a total of C classes (global labels). Denote by μ the meta-distribution sampling distributions (a.k.a. tasks) ρ, from which we sample support and query sets (S, Q) for each task. Each task distribution ρ is associated with $k \leq C$ class labels $y_\rho^{(1)}, \dots, y_\rho^{(k)} \in \{1, \dots, C\}$. Denote by $\rho_Y = \{y_\rho^{(1)}, \dots, y_\rho^{(k)}\}$ the corresponding subset of $\{1, \dots, C\}$. Given a matrix $W \in \mathbb{R}^{C \times m}$ and a vector $Y \in \{1, \dots, C\}^k$ of indices, we denote by $W[Y] = W[\rho_Y] \in \mathbb{R}^{k \times m}$ the submatrix of W obtained by selecting the rows corresponding to the unique indices ρ_Y in Y. Lastly, Given a dataset $D = (x_i, y_i)_{i=1}^n$ we denote by $D_Y \in \{1, \dots, C\}^n$ the vector with entries corresponding to the labels y_i.

We also define the expected error incurred by a meta-learning algorithm solving (4)

$$\mathcal{E}(\theta, \mu) = \mathbb{E}_{\rho \sim \mu} \mathbb{E}_{(S,Q)\sim\rho} \mathcal{L}(w(g_\theta(S)), g_\theta(Q)). \tag{9}$$

This is the meta-learning risk incurred by a meta-parameter θ, namely the error incurred by training the classifier via $w(g_\theta(S))$ (e.g., (5)) and testing it on the query set $g_\theta(Q)$, averaged over (S, Q) pairs sampled from tasks ρ, which in turn are sampled from meta-distribution μ. The risk is the ideal error we wish to minimize.

B. Global Label Selection (GLS)

We start our analysis by introducing a special FSL scenario that will be useful for understanding the relation between pre-training and meta-learning. To this end, we assume in this scenario that global labels are available to the model.

Given the access to global labels, we can design a new algorithm that learns a single global multi-class linear classifier W at the meta-level (i.e., shared across all tasks), and simply selects the required rows $W[S_Y]$ when tackling a task. More formally, we can define a special base learner called *global label selector (GLS)* such that

$$\text{Alg}((W, \theta), S) = \text{GLS}(W, \theta, S) = W[S_Y] g_\theta(\cdot).$$

Illustrated in Fig. 1(b), this "algorithm" does not solve an optimization problem on the support set S, but only selects the subset of rows of W corresponding to the classes present in S as the task-specific classifier.

Since W and θ are now both shared across all tasks, we may learn them jointly by minimizing the following

$$\mathbb{E}_{(S,Q)\in\mathcal{T}} \mathcal{L}(W[S_Y] g_\theta(\cdot), Q), \tag{10}$$

over both W and θ. This strategy, to which we refer as *meta-GLS*, learns both the representation and linear classifier at the meta-level, with the sole task-specific adaptation process being the selection of columns of W using the global labels.

GLS Finds a Good Meta-representation: Learning a global W shared among multiple tasks (rather than having each classifier $w(g_\theta(S))$ accessing exclusively the tasks' training data), can be very advantageous for generalization. This is evident when the (global) classes are separable for a meta-representation g_θ. Let

$$\mathcal{E}_{\text{GLS}}(W, \theta, \mu) = \mathbb{E}_{\rho\sim\mu} \mathbb{E}_{(S,Q)\sim\rho} \mathcal{L}(W[S_Y] g_\theta(\cdot), Q), \tag{11}$$

denote the expected meta-GLS risk incurred by the minimizer of (10). Then, for any inner algorithm $w(\cdot)$, we have

$$\min_W \mathcal{E}_{\text{GLS}}(W, \theta, \mu) \leq \mathcal{E}(\theta, \mu), \tag{12}$$

namely that, for any given representation g_θ, finding a global classifier W for all classes is more favorable than solving each task in isolation. In other words, *solving meta-GLS provides a good representation $g_\theta(\cdot)$ for standard meta-learning problem.*

C. Pre-Training and GLS

Existing works such as [15], [71] demonstrates that pre-training offers a robust alternative to learn the meta-representation $g_\theta(\cdot)$. We will show that GLS is related to pre-training, under some mild assumptions.

Assumption 1: The meta-distribution μ samples tasks ρ. Sampling from each ρ is performed as follows:

1) For each $j \in \{1, \dots, k\}$ and class $y_\rho^{(j)} \in \rho_Y$, we sample n examples $x_1^{(j)}, \dots, x_n^{(j)}$ i.i.d. from a conditional distribution $\pi(x|y = y_\rho^{(j)})$ shared across all tasks. All generated pairs are collected in the support set $S = (x_i^{(j)}, y_\rho^{(j)})_{i,j=1}^{n,k}$.
2) The query set Q is generated by sampling m points i.i.d. from $\pi_\rho(x, y)$, namely $Q \sim \pi_\rho^m$ with

$$\pi_\rho(x, y) = \pi(x|y)\text{Unif}_\rho(y) \tag{13}$$

and Unif_ρ the uniform distribution over the labels in ρ_Y.

In essence, Assumption 1 characterizes the standard process of constructing meta-training tasks for FSL, typically adopted to build pre-training datasets in practice. In particular, let $\pi_\mu(x, y)$ be the marginal probability of observing (x, y) in the meta-training tasks, i.e., first sampling a task ρ from μ, followed by sampling a class y uniformly by $\text{Unif}_\rho(\cdot)$ and finally x by $\pi(\cdot|y)$. It then follows that sampling a dataset D_{global} from π_μ is equivalent to sample a meta-training set $\mathcal{T}$ from μ and flatten it into $D(\mathcal{T})$ according to the pre-training procedure described in (7).

We can therefore introduce the *(global) multi-class classification* risk associated to π_μ

$$\mathcal{L}(Wg_\theta(\cdot), \pi_\mu) = \mathbb{E}_{(x,y)\sim\pi_\mu} \ell_{\text{ce}}(Wg_\theta(x), y). \tag{14}$$

The above risk can be seen as the ideal objective for the pre-training estimator in (8). In addition, the following result relates pre-training to meta-GLS.

Theorem 1: Under Assumption 1, let $\pi_\mu(x, y)$ be the marginal distribution of observing (x, y) in the meta-training set. Then, for any (global) classifier W,

$$\mathcal{E}_{\text{GLS}}(W, \theta, \mu) \le \mathcal{L}(W, \theta, \pi_\mu). \tag{15}$$

Moreover, if the global classes are separable,

$$\min_{W,\theta} \mathcal{E}_{\text{GLS}}(W, \theta, \mu) = \min_{W,\theta} \mathcal{L}(W, \theta, \pi_\mu), \tag{16}$$

The result shows that the GLS error is upper bounded by the global multi-class classification error. Hence, minimizing the global multi-class classification error also indirectly minimizes the meta-learning risk. This implies that *pre-training implicitly learns a meta-representation suitable for FSL.*

D. Generalization Properties

Theorem 1 shows that under the class-separability assumption, pre-training is equivalent to performing meta-GLS. We now study which of the two approaches is more sample-efficient from a generalization perspective.

Let $(W_T^{\text{GLS}}, \theta_T^{\text{GLS}})$ denote the meta-parameters learned by an algorithm minimizing (10) over a dataset $\mathcal{T}$ comprising of T separate tasks. Applying standard results from statistical learning theory, we can obtain excess risk bounds characterizing the quality of θ_T's predictions in terms of the number T of tasks the algorithm has observed in training. For instance, following [65, Chapter 26] we have that in expectation with respect to sampling $\mathcal{T}$

$$\begin{aligned}\mathbb{E}_{\mathcal{T}}[\mathcal{E}_{\text{GLS}}(W_T^{\text{GLS}}, \theta_T^{\text{GLS}}, \mu)] &\le \min_{(W,\theta)\in\Omega} \mathcal{E}_{\text{GLS}}(W, \theta, \mu) + 2L_{\text{GLS}}\mathfrak{R}_T(\Omega) \\ &\le \min_{(W,\theta)\in\Omega} \mathcal{E}_{\text{GLS}}(W, \theta, \mu) + \frac{2L_{\text{GLS}}C_\Omega}{\sqrt{T}}\end{aligned}$$

where L_{GLS} denotes the Lipschitz constant of $\mathcal{E}_{\text{GLS}}$, while $\Omega \subset \mathbb{R}^{m\times C} \times \Theta$ is the space of hypotheses for the multi-class classifier $Wg_\theta(\cdot)$. Here, $\mathfrak{R}_T(\Omega)$ is the Rademacher complexity of Ω [65], which measures the overall potential expressivity of an estimator that can be trained over them. For neural networks, [22] showed that $\mathfrak{R}_T(\Omega)$ may be further bounded by $\mathfrak{R}_T(\Omega) \le C_\Omega/\sqrt{T}$, where C_Ω is a constant depending on the specific neural architecture, with deeper networks having a larger constant. The bound indicates that the risk incurred by GLS becomes closer to that of the ideal meta-parameters as the number of observed tasks T grows.

We can apply the same Rademacher-based bounds to (14) and the pre-training estimator from (8), obtaining that in expectation with respect to sampling $\mathcal{T}$

$$\mathbb{E}_{\mathcal{T}}[\mathcal{L}(W_N^{\text{pre}}, \theta_N^{\text{pre}}, \pi_\mu)] \le \min_{(W,\theta)\in\Omega} \mathcal{L}(W, \theta, \pi_\mu) + \frac{2L_{\text{pre}}C_\Omega}{\sqrt{N}}$$

where N is the number of samples in D_{global} and L_{pre} is the Lipschitz constant of the global multi-class classification risk. By combining the above bound with the result from Theorem 1 we conclude that

$$\mathbb{E}_{\mathcal{T}}[\,\mathcal{E}_{\text{GLS}}(W_N^{\text{pre}}, \theta_N^{\text{pre}}, \mu)] \le \min_{W,\theta} \mathcal{E}_{\text{GLS}}(W, \theta, \mu) + \frac{2L_{\text{pre}}C_\Omega}{\sqrt{N}},$$

which is an excess risk bound analogous to that obtained for meta-GLS. The key difference is that the bound above depends on the number N of total samples in D_{global}, rather that the total number T of tasks.

Comparing the rates of meta-GLS and the pre-training estimator, we observe that typically $N \gg T$ (for instance $N = nT$ when each task has the same number of n samples). Additionally, since L_{pre} is comparable or smaller than L_{GLS} (see Appendix B, available online), we conclude that

Given exactly the same data ($\mathcal{T}$ for meta-GLS and $D(\mathcal{T})$ for pre-training), pre-training achieves a much smaller error than meta-GLS.

For instance, in the case of a 5-way-5-shot FSL problem, pre-training improves upon the meta-GLS bound on excess risk by a factor of $\sqrt{N/T} = \sqrt{n} = \sqrt{100} = 10$.

Given the relation between GLS and standard meta-learning that we highlighted in Section III-B, our analysis provides a strong theoretical argument in favor of adopting pre-training in meta-learning settings. To our knowledge, this is a novel and surprising result.

E. Connection to Conditional Meta-Learning

More generally, we observe that GLS is also an instance of conditional meta-learning: the global labels of the task provide additional contextual information about the task to facilitate model learning. Global labels directly reveal how tasks relate to one another and in particular if any classes to be learned are shared across tasks. GLS thus simply map global labels of

tasks to task classifiers via $W[S_Y]$. In contrast, unconditional approaches (e.g., R2D2 [6], ProtoNet [48]) learn classifiers by minimizing some loss over support sets, losing out on the access to the contextual information provided by global labels.

In addition to our result, [11], [12] also proved that conditional meta-learning is advantageous over the unconditional formulation by incurring a smaller excess risk, especially when the meta-distribution of tasks is organized into distant clusters. We refer readers to the original papers for a detailed discussion. In practice, global labels provide clustering of task samples for free and improve regularization by enforcing each cluster (denoted by global label y_ρ^j) to share classifier parameters $W[y_\rho^j]$ across all tasks. This provides further explanation to why pre-training yields a robust meta-representation with strong generalization performance.

F. Leveraging Pre-Training in Practice

The goal of meta-learning is to generalize to novel classes unseen during training. Therefore, practical FSL applications assume meta-testing and meta-training distributions to share no class labels. To apply our analysis in Section III-D to these settings, we may follow the theoretical approach in [13] and assume that meta-training and meta-testing classes share a common representation. The assumption is reasonable since extensive empirical evidences demonstrate that pre-trained representation on meta-training set is robust for directly classifying novel classes [15], [71]. To prevent overfitting on meta-training set and ensure a robust represntation for meta-testing, well-established techniques e.g., [3], [71], [77] include imposing ℓ_2 regularization during pre-training (see weight decay in Appendix C.2, available online) and early stopping by performing meta validation.

While pre-training might offer a powerful initial representation θ – as highlighted by our analysis in Section III-D – it may be advisable to further improve θ. One general strategy is to fine-tune θ by directly optimizing (4) using the desired classifier to tackle novel classes [e.g. [60], [72], [84], [85]. This strategy is known as *meta fine-tuning*. A different approach is based on a transfer learning perspective. Specifically, [33], [38], [66] showed that careful task-specific fine-tuning (e.g., limiting the number of learnable parameters) from a pre-trained representation achieves robust generalization performance, even in FSL settings. We investigate both strategies in our experiments.

IV. Methods

In this section, we propose three practical algorithms motivated by our theoretical analysis. In Section IV-A, we introduce an augmentation procedure for pre-training to further improve representation learning in image-based tasks. In Section IV-B, we tackle the scenario where global labels are absent by automatically inferring a notion of global labels. Lastly, we introduce a meta fine-tuning procedure in Section IV-C to investigate how much meta-learning could improve the pre-trained representation.

A. Augmented Pre-Training for Image-Based Tasks

In general, pre-training is a standard process with well-studied techniques for improving the final learned representation. Many of these techniques, including data augmentation [8], auxiliary losses [45] and model distillation [71], are also effective for FSL (i.e., the learned representation is suitable for novel classes during meta-testing). In particular, we may interpret data augmentation techniques as increasing N in the bounds for the pre-training estimator outlined in Section III-D, thus improving the error incurred by pre-training and consequently the learned representation g_θ.

In addition to standard augmentations (e.g., random cropping and color jittering) investigated in [8], we further propose an augmented procedure for pre-training via image rotation. For every class y_i in the original dataset, we create three additional classes by rotating all images of class y_i by $r \in \{90°, 180°, 270°\}$ respectively. All rotations are multiples of $90°$ such that they can be implemented by basic operations efficiently (e.g., flip and transpose) and prevent pre-training from learning any trivial features from visual artifacts produced by arbitrary rotations [20]. Pre-training is then performed normally on the augmented dataset.

The augmented dataset quadruples the number of samples and classes compared to the original dataset. According to our analysis from Section III-D, pre-training on the augmented dataset may yield a more robust representation. Further, we also hypothesize that the quality of the representation also depends on the number of classes available in the pre-training dataset, since classifying more classes requires learning increasingly discriminating representations. Our experiments show that 1) augmented pre-training consistently outperforms the standard one, and 2) quality of the learned representation depends on both the dataset size and the number of classes available for training.

B. Meta Label Learning

The ability to exploit pre-training crucially depends on access to global labels. However, as discussed in Section I, *global labels may be inaccessible in practical applications*. For instance when meta-training tasks are collected and annotated independently. Additionally, tasks may have conflicting labels over similar data points based on different task requirements – a setting illustrated by our experiments in Section V-D. Therefore in some applications, global labels are ill-defined, and pre-training is not directly applicable.

To tackle this problem, we consider the more general setting where only local labels from each task are known. This setting is also the one originally adopted by most earlier works in meta-learning [e.g., [6], [17], [36], [39], [68], [75]. In the local label setting, we propose Meta Label Learning (MeLa), which automatically infer a notion of latent global labels across tasks. The inferred labels enable pre-training and thus bridge the gap between the experiment settings with and without global labels. We stress that our proposed method *does not* replace standard pre-training with global labels, but rather provides

Algorithm 1: MeLa.

Input: meta-training set $\mathcal{T} = \{S_t, Q_t\}_{t=1}^{T}$
$g_\theta^{\text{sim}} = \text{argmin}_{g_\theta} \mathbb{E}_{(S,Q)\in\mathcal{T}}[\mathcal{L}(w(g_\theta(S)), g_\theta(Q))]$
Global clusters $G = \text{LearnLabeler}(\mathcal{T}, g_\theta^{\text{sim}})$
$g_\theta^{\text{pre}} = \text{Pretrain}(D(\mathcal{T}), G)$
$g_\theta^{*} = \text{MetaFinetune}(G, \mathcal{T}, g_\theta^{\text{pre}})$
Return g_θ^{*}

a way to still benefit from such a strategy when they are absent.

Algorithm 1 outlines the general approach for learning a few-shot model using MeLa: we first meta-learn an initial representation g_θ^{sim}; Second, we cluster all task samples using g_θ^{sim} as a feature map while enforcing local task constraints. The learned clusters are returned as inferred global labels. Using the inferred labels, we can apply pre-training to obtain g_θ^{pre}, which may be further fine-tuned to derive the final few-shot model g_θ^{*}. We present in Section IV-C a simple yet effective meta fine-tuning procedure.

For learning g_θ^{sim}, we directly optimize (4) using ridge regression (5) as the base learner. We use ridge regression for its computational efficiency and good performance. Using g_θ^{sim} as a base for a similarity measure, the labeling algorithm takes as input the meta-training set and outputs a set of clusters as global labels. The algorithm consists of a clustering routine for sample assignment and centroid updates and a pruning routine for merging small clusters.

Clustering: The clustering routine leverages local labels for assigning task samples to appropriate global clusters and enforcing task constraints. We observe that for any task, the local labels describe two constraints: 1) samples sharing a local label must be assigned to the same global cluster, while 2) samples with different local labels must not share the same global cluster. To meet constraint 1, we assign all samples $\{x_i^{(j)}\}_{i=1}^{n}$ of class $y_\rho^{(j)}$ to a single global cluster by

$$v^* = \underset{v\in\{1,\dots,V\}}{\text{argmin}} \left\| \frac{1}{n}\sum_{i=1}^{n} g_\theta^{\text{sim}}(x_i^{(j)}) - g_v \right\|^2, \tag{17}$$

with V being the current number of centroids.

We apply (17) to all classes $y_\rho^{(1)}, \dots, y_\rho^{(k)}$ within a task. If multiple local classes map to the same global label, we simply discard the task to meet constraint 2. Otherwise, we proceed to update the centroid g_{v^*} and sample count N_{v^*} for the matched clusters using

$$g_{v^*} \leftarrow \frac{N_{v^*} g_{v^*} + \sum_{i=1}^{n} g_\theta^{\text{sim}}(x_i)}{N_{v^*} + n},$$
$$N_{v^*} \leftarrow N_{v^*} + n, \tag{18}$$

Pruning. We also introduce a strategy for pruning small clusters. We model the sample count of each cluster as a binomial distribution $N_v \propto B(T, p)$. We set $p = \frac{1}{V}$, assuming that each cluster is equally likely to be matched by a local class of samples. Any cluster with a sample count below the following threshold is discarded,

$$N_v < \bar{N_v} - q\sqrt{\text{Var}(N_v)} \tag{19}$$

where $\bar{N_v}$ is the expectation of N_v, $\text{Var}(N_v)$ the variance, and q a hyper-parameter controlling how aggressive the pruning is.

Algorithm 2: LearnLabeler.

Input: embedding model g_θ^{sim}, meta-training set $\mathcal{T} = \{S_t, Q_t\}_{t=1}^{T}$, number of classes in a task k
Initialization: sample tasks from $\mathcal{T}$ to initialize clusters $G = \{g_v\}_{v=1}^{V}$,
While $|G|$ has not converged:
$N_v = 1$ for each $g_v \in G$
For $(S, Q) \in \mathcal{T}$:
Match $S \cup Q$ to its centroids $M = \{g_q\}_{q=1}^{K}$ using (17)
If M has k unique clusters
Update centroid g_q for each $g_q \in M$ via (18)
$G \leftarrow \{g_v | g_v \in G, N_v \geq$ threshold in (19)$\}$
Return G

Algorithm 2 outlines the full labeling algorithm. We first initialize a large number of clusters by setting their centroids with mean class embeddings from random classes in $\mathcal{T}$. For V initial clusters, $\lceil \frac{V}{k} \rceil$ tasks are needed since each task contains k classes and could initialize as many clusters. The algorithm then alternates between clustering and pruning to refine the clusters and estimate the number of clusters jointly. The algorithm terminates and returns the current clusters G when the number of clusters does not change from the previous iteration. Using clusters G, local classes from the meta-training set can be assigned global labels with nearest neighbor matching using (17). For tasks that fail to map to k unique global labels, we simply exclude them from the pre-training process.

The key difference between Algorithm 2 and the classical K-means algorithm [42] is that the proposed clustering algorithm exploits local information to guide the clustering process, while K-means algorithm is fully unsupervised. We will show in the experiments that enforcing local constraints is necessary for learning robust meta-representation.

Algorithm 2 also indirectly highlights how global labels, if available, offer valuable information about meta-training set. In addition to revealing precisely how input samples relate to one another across tasks, global labels provide an overview of meta-training set, including the desired number of clusters and their sizes. In contrast, Algorithm 2 needs to estimate both properties when only local labels are given.

Time Complexity: The time complexity of training MeLa is dominated by the computational cost of pretraining, accounting for over 70% of the overall running time. From our benchmarks, the time complexity of MeLa is comparable to those of the current state-of-the-art methods based on pre-training e.g., [80], [84] and significantly more efficient than methods relying on complex base learners e.g., [85]. We refer to Section C.3 for a more detailed discussion and comparison with [80], [84], [85].

When global labels are not available MeLa requires performing an additional inference step to estimate them. While this stage accounts for around 20% of the total running time, we observe in Section V that it provides a significant performance boost compared to FSL methods not utilizing pre-training, which are the only applicable ones in the absence of global labels.

C. Meta Fine-Tuning

As discussed in Section III-F, while pre-training already yields a robust meta-representation for FSL, it is desirable to adapt the pre-trained representation such that the new meta-representation better matches the base learner intended for novel classes. We call this additional training *meta fine-tuning*, which is adopted by several state-of-the-art FSL models [37], [77], [84], [85].

For meta fine-tuning, existing works suggest that model performance depends crucially on preserving the pre-trained representation. In particular, [37], [62], [77] all keep the pre-trained representation fixed, and only learn a relatively simple transformation on top for the new base learners. Additionally, [21] showed that meta fine-tuning the entire representation model lead to worse performance compared to standard meta-learning, negating the advantages of pre-training completely.

We thus present a simple residual architecture that preserves the pre-trained embeddings and allows adaptation for the new base learner. Formally, we consider the following parameterization for a fine-tuned meta-learned embedding g^*_θ,

$$g^*_\theta(x) = g^{\text{pre}}_\theta(x) + h(g^{\text{pre}}_\theta(x)) \tag{20}$$

where g^{pre}_θ is the pre-trained representation and h a learnable function (e.g., a small fully connected network). We again use (5) as the base learner and optimizes (4) directly. Our experiments show that the proposed fine-tuning process achieves results competitive with more sophisticated base learners, indicating that the pre-trained representation is the predominant contributor to good test performance.

V. Experiments

We evaluate MeLa on various benchmark datasets and compare it with existing methods. The experiments are designed to address the following questions:

- How does MeLa compare to existing methods for generalization performance? We also introduce the more challenging GFSL setting in Section V-B.
- How do different model components (e.g., pre-training, meta fine-tuning) contribute to generalization performance?
- Does MeLa learn meaningful clusters? Can MeLa handle conflicting task labels?
- How can we improve the quality of the pre-trained representation?
- How robust is MeLa to hyper-parameter choices?

A. Benchmark Datasets

Mini/tiered-ImageNet: [57], [75] has become default benchmark for FSL. Both datasets are subsets of ImageNet [61] with *mini*ImageNet having 60 K images over 100 classes, and *tiered*ImageNet having 779 K images over 608 classes. Following previous works, we report performance on 1- and 5-shot settings, using 5-way classification tasks.

Variants of mini/tiered-ImageNet: We introduce several variants of mini/tiered-ImageNet to better understand MeLa and more broadly the impacts of dataset configuration on pre-training. Specifically, we create mini-60 that consists of 640 classes and 60 samples per class. Mini-60 contains the same number of samples as *mini*ImageNet, though with more classes and fewer samples per class. Mini-60 keeps the same meta-test set as *mini*ImageNet to ensure a fair comparison of test performance of model trained on each dataset in turn. We designed mini-60 to investigate the behavior of MeLa when encountering a dataset with a high number of base classes and low number of samples per base class. We also use mini-60 to explore how data diversity present in the training data affects the learned representation. Analogous to mini-60, we also introduce tiered-780 as a variant to *tiered*ImageNet, where we take the total number of samples in *tiered*ImageNet and calculate the number of samples over the full 1000 ImageNet classes, excluding those used in the meta-test set of *tiered*ImageNet.

Meta-Dataset: [72] is a meta-learning classification benchmark combining 10 widely used datasets: ILSVRC-2012 (ImageNet) [61], Omniglot [35], Aircraft [44], CUB200 [79], Describable Textures (DTD) [10], QuickDraw [32], Fungi [64], VGG Flower (Flower) [49], Traffic Signs [28] and MSCOCO [40]. We use Meta-Dataset to construct several challenging experiment scenarios, including learning a unified model for multiple domains and learning from tasks with conflicting labels.

B. Experiment Settings

The standard FSL setting [5], [17], [68], [80], [84] assumes that a meta-distribution of tasks is available for training. This translates to meta-learners having access to an exponential number of tasks synthetically generated from the underlying dataset, a scenario unrealistic for practical applications. Recent works additionally assume access to global labels in order to leverage pre-training, in contrast with earlier methods that assume access to only local labels. We will highlight such differences when comparing different methods.

Generalized Few-Shot Learning (GFSL) Setting: We introduce a more challenging and realistic FSL setting. Specifically, we only allow access to local labels, since global ones may be inaccessible or ill-defined. In addition, we employ a *no-replacement* sampling scheme when synthetically generating tasks from the underlying dataset.[1] This sampling process limits the meta-training set to a fixed-size, a standard assumption for most machine learning problems. The fixed size also enables

[1] For instance, *mini*ImageNet (38400 training samples) will be randomly split into around 380 tasks of 100 samples.

TABLE I
TEST ACCURACY OF META-LEARNING MODELS ON *mini*IMAGENET AND *tiered*IMAGENET

	*mini*IMAGENET		*tiered*IMAGENET	
	1-shot	5-shot	1-shot	5-shot
Global Labels				
Simple CNAPS [5]	$53.2 \pm -$	$70.8 \pm -$	$63.0 \pm -$	$80.0 \pm -$
LEO [62]	61.7 ± 0.7	77.6 ± 0.4	66.3 ± 0.7	81.4 ± 0.6
TASML [77]	62.0 ± 0.5	78.2 ± 0.5	66.4 ± 0.4	82.6 ± 0.3
RFS [71]	62.0 ± 0.4	79.6 ± 0.3	69.4 ± 0.5	84.4 ± 0.3
ProtoNet (with pre-train) [80]	62.4 ± 0.2	80.5 ± 0.1	68.2 ± 0.2	84.0 ± 0.3
Meta-Baseline [9]	63.2 ± 0.2	79.3 ± 0.2	68.6 ± 0.3	83.7 ± 0.2
FEAT [84]	$\mathbf{66.7 \pm 0.2}$	82.0 ± 0.1	70.8 ± 0.2	84.8 ± 0.2
FRN [80]	66.4 ± 0.2	$\mathbf{82.8 \pm 0.1}$	$\mathbf{71.2 \pm 0.2}$	$\mathbf{86.0 \pm 0.2}$
DeepEMD [85]	65.9 ± 0.8	82.4 ± 0.6	$\mathbf{71.2 \pm 0.9}$	$\mathbf{86.0 \pm 0.6}$
Local Labels				
MAML [17]	48.7 ± 1.8	63.1 ± 0.9	51.7 ± 1.8	70.3 ± 0.8
ProtoNet [68]	49.4 ± 0.8	68.2 ± 0.7	53.3 ± 0.9	72.7 ± 0.7
R2D2 [6]	51.9 ± 0.2	68.7 ± 0.2	65.5 ± 0.6	80.2 ± 0.4
MetaOptNet [36]	62.6 ± 0.6	78.6 ± 0.5	66.0 ± 0.7	81.5 ± 0.6
Shot-free [56]	$59.0 \pm \text{n/a}$	$77.6 \pm \text{n/a}$	$63.5 \pm \text{n/a}$	$82.6 \pm \text{n/a}$
MeLa (pre-train only)	64.5 ± 0.4	81.5 ± 0.3	69.5 ± 0.5	84.3 ± 0.3
MeLa	$\mathbf{65.8 \pm 0.4}$	$\mathbf{82.8 \pm 0.3}$	$\mathbf{70.5 \pm 0.5}$	$\mathbf{85.9 \pm 0.3}$

us to evaluates the sample efficiency of different methods. Second, no-replacement sampling prevents MeLa and other meta-learners from trivially learning task relations, a key objective of meta-learning, by matching same samples across tasks. For instance, an identical sample appearing in multiple tasks would allow MeLa to trivially cluster local classes. Lastly, the sampling process reflects any class imbalance in the underlying dataset, which might present a more challenging problem.

C. Performance Comparison in Standard Setting

We compare MeLa to a diverse group of existing methods on mini- and *tiered*IMAGENET in Table I. We separate the methods into those requiring global labels and those that do not. We note that the two groups of methods are not directly comparable since global labels provides a significant advantage to meta-learners as discussed previously. The method groupings are intended to demonstrate the effect of pre-training on generalization performance. Lastly, bold values in all tables indicate the best performing models.

Table I clearly shows that "global-labels" methods leveraging pre-training generally outperform "local-labels" methods except MeLa. We highlight that the re-implementation of ProtoNet in [80] benefits greatly from pre-training, outperforming the original by over 10% across the two datasets. Similarly, while RFS and R2D2 both learn a fixed representation and only adapt the classifier based on each task, RFS's pre-trained representation clearly outperforms R2D2's meta-learned representation. We further note that state-of-the-art methods such as DeepEMD and FEAT are heavily reliant on pre-training and performs drastically worse in GFSL setting, as we will discuss in Section V-D.

In the local-labels category, MeLa outperforms existing methods thanks to its ability to still exploit pre-training using the inferred labels. MeLa achieves about 4% improvement over the next best method in all settings. Across both categories, MeLa obtains performance competitive to state-of-the-art methods such as FRN, FEAT and DeepEMD despite having no access to global labels. This indicates that MeLa is able to infer meaningful clusters to substitute global labels and obtains performance similar to methods having access to global labels. We will provide further quantitative results on the clustering algorithm in Section V-G. Lastly, we note that MeLa also outperforms several methods from the "global-label" category, such as RFS and Meta-Baseline. We attribute MeLa's better performance to more robust representation via augmented pre-training and our formulation for meta fine-tuning. In particular, we explicitly preserve the pre-trained representation using residual connections, in contrast to meta fine-tuning the entire representation model as in ProtoNet and Meta-Baseline. Consistent with [21], the results suggest that meta fine-tuning the entire representation model could negate the advantages of pre-training shown in our theoretical analysis.

D. Performance Comparison in Generalized Setting

We evaluate a representative set of few-shot learners under GFSL. For this setting, we introduce two new experimental scenario using Meta-Dataset to simulate task heterogeneity.

In the first scenario, we construct the meta-training set from Aircraft, CUB and Flower, which we simply denote as "Mixed". Tasks are sampled independently from one of the three datasets. For meta-testing, we sample 1500 tasks from each dataset and report the average accuracy. The chosen datasets are intended for fine-grained classification in aircraft models, bird species and flower species respectively. Thus the meta-training tasks share the broad objective of fine-grained classification, but are sampled from three distinct domains. A key challenge of this scenario is to learn a unified model across multiple domains, without any explicit knowledge about them or the global labels.

Table II show that MeLa outperforms all baselines under GFSL setting. In particular, MeLa achieves a large margin of

TABLE II
TEST ACCURACY ON AIRCRAFT, CUB AND VGG FLOWER (MIXED DATASET)

	Aircraft		CUB		VGG Flower		Average	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
ProtoNet [68]	35.1 ± 0.4	51.0 ± 0.5	32.7 ± 0.4	46.4 ± 0.5	56.7 ± 0.5	73.8 ± 0.4	41.7 ± 0.7	57.5 ± 0.7
MatchNet [75]	31.4 ± 0.4	39.4 ± 0.5	42.7 ± 0.5	54.1 ± 0.5	62.5 ± 0.5	70.0 ± 0.1	45.7 ± 0.1	54.5 ± 0.1
R2D2 [6]	67.7 ± 0.6	82.8 ± 0.4	53.8 ± 0.5	69.2 ± 0.5	65.4 ± 0.5	83.3 ± 0.3	61.9 ± 0.5	78.6 ± 0.4
DeepEMD [85]	34.7 ± 0.7	47.8 ± 1.4	39.3 ± 0.7	52.1 ± 1.4	61.3 ± 0.9	74.5 ± 1.4	45.1 ± 0.7	58.1 ± 1.2
FEAT [84]	61.7 ± 0.6	75.8 ± 0.5	59.6 ± 0.6	73.1 ± 0.5	62.9 ± 0.6	76.0 ± 0.4	60.9 ± 0.7	75.0 ± 0.5
FRN [80]	60.7 ± 0.7	77.6 ± 0.5	61.9 ± 0.7	77.7 ± 0.5	65.2 ± 0.6	81.2 ± 0.5	63.1 ± 0.7	79.7 ± 0.5
MeLa	$\mathbf{78.2 \pm 0.5}$	$\mathbf{89.5 \pm 0.3}$	$\mathbf{73.8 \pm 0.6}$	$\mathbf{88.7 \pm 0.3}$	$\mathbf{76.6 \pm 0.4}$	$\mathbf{91.5 \pm 0.2}$	$\mathbf{76.2 \pm 0.3}$	$\mathbf{89.9 \pm 0.2}$

A single model is trained for each method over all tasks.

TABLE III
TEST ACCURACY ON H-AIRCRAFT IN THE GENERALIZED SETTING

	1-shot	5-shot
ProtoNet [68]	47.8 ± 0.5	66.8 ± 0.5
MatchNet [75]	65.6 ± 0.2	78.7 ± 0.2
R2D2	75.1 ± 0.3	86.4 ± 0.2
DeepEMD [85]	51.3 ± 0.5	65.6 ± 0.8
FEAT [84]	77.6 ± 0.6	87.3 ± 0.4
FRN [80]	81.9 ± 0.4	91.0 ± 0.2
MeLa	$\mathbf{84.8 \pm 0.3}$	$\mathbf{92.9 \pm 0.2}$
Oracle	84.4 ± 0.3	93.1 ± 0.2

10% improvement over the baselines, including state-of-the-art models FEAT, FRN and DeepEMD, which performed competitively against MeLa in Table I. In particular, FEAT and DeepEMD performed noticeably worse, indicating the methods' reliance on pre-trained representation and the difficulty of meta-learning from scratch with complex base learners. FRN outperforms FEAT and DeepEMD, as it is designed to also work without pre-training.

In the second scenario, we consider meta-training tasks with heterogeneous objectives, leading to conflicting task-labels and consequently ill-defined global labels. For the Aircraft dataset, each sample from the base dataset has three labels associated with it, including variant, model and manufacturer[2] that form a hierarchy. We sample tasks based on each of the three labels and creates a meta-training set containing three different task objectives: classifying fine-grained differences between model variants, classifying different airplanes, and classifying different airplane manufacturers. To differentiate from the original dataset, we refer to this meta-training set as H-Aircraft. The training data is particularly challenging given the competing goals across different tasks: a learner is required to recognize fine-grained differences between airplane variants, while being able to identify general similarities within the same manufacturer. The meta-training data also reflects the class imbalance of underlying dataset, with samples from Boeing and Airbus over-represented.

Table III shows that MeLa outperforms all baselines for H-Aircraft. To approximate the oracle performance when ground truth labels were given, we optimize a supervised semantic softmax loss [59] over the hierarchical labels. Specifically, we train the (approximate) oracle to minimize a multi-task objective combining individual cross entropy losses over the three labels. MeLa performs competitively against the oracle, indicating the robustness of the proposed labeling algorithm in handling ill-defined labels and class imbalance.

The experimental results suggest that MeLa performs robustly in both the standard and GFSL settings. In contrast, baseline methods perform noticeably worse in the latter, due to the absence of pre-training and limited training data.

Connection to theoretical results: We comment on the empirical results so far in relation to our theoretical analysis. The empirical results strongly indicate that pre-training produces robust meta-representations for FSL by exploiting contextual information from global labels. This is consistent with our observation that pre-training would achieve a smaller error than its meta-learning counterpart. On the other hand, the results also validate our hypothesis that the pre-trained representation can be further improved, since the pre-trained representation is not explicitly optimized for handling novel classes. In particular, FEAT, FRN, DeepEMD and MeLa all outperform the pre-trained representation from [71] by further adapting it.

E. Performance Comparison on Meta-Dataset

We further evaluate MeLa on the full Meta-Dataset to assess our method's generalization performance. We adopt the experiment setting of training on ImageNet only and testing on all meta-test sets [72], to clearly evaluate out-of-distribution generalization. We note that state-of-the-art methods [e.g.[38], [63], [73] on Meta-Dataset are heavily reliant on pre-training with global labels, while MeLa only has access to a collection of FSL tasks and has to infer such labels. In Table IV, we compare MeLa with state-of-the-art methods including fine tuning [72], ALFA+fo-Proto-MAML [72], BOHB [63], FLUTE [73] and TSA [38].

The results show that MeLa is able to effectively infer meaningful global labels and achieve robust generalization to novel datasets, achieving an average accuracy of 68.5%. Despite not being given global labels for pre-training, MeLa only trails behind TSA while outperforming other methods. In addition, we note that the task-specific tuning adopted by TSA is orthogonal – but compatible – to MeLa: by combining MeLa with TSA (see Section C.2 for details) we are able to further improve our generalization performance, outperforming the original TSA approach on 10 out of the 13 meta-test sets (Table IV last column). These results further demonstrate the robustness of MeLa in learning robust representations over a large number of FSL tasks, and the efficacy of task-specific fine-tuning in improving generalization of novel tasks.

[2]E.g., "Boeing 737-300" indicates manufacturer, model, and variant.

TABLE IV
TEST ACCURACY ON META-DATASET, TRAINING ON IMAGENET ONLY, USING RESNET-18 FOR ALL MODELS

Test Dataset	Finetune [72]	fo-Proto-MAML [72]	BOHB [63]	FLUTE [73]	Meta-Baseline [9]	TSA [38]	MeLa	MeLa+TSA
ImageNet	45.8 ± 1.1	52.8 ± 1.1	51.9 ± 1.1	46.9 ± 1.1	$59.2 \pm -$	57.4 ± 1.0	59.3 ± 1.1	$\mathbf{61.3} \pm 1.1$
Omniglot	60.9 ± 1.6	61.9 ± 1.5	67.6 ± 1.2	61.6 ± 1.4	$69.1 \pm -$	74.2 ± 1.2	66.2 ± 1.4	$\mathbf{74.8} \pm 1.4$
Aircraft	68.7 ± 1.3	63.4 ± 1.1	54.1 ± 0.9	48.5 ± 1.0	$54.1 \pm -$	66.1 ± 1.0	67.9 ± 1.0	$\mathbf{80.7} \pm 1.1$
Birds	57.3 ± 1.3	69.8 ± 1.1	70.7 ± 0.9	47.9 ± 1.0	$77.3 \pm -$	73.9 ± 0.9	78.7 ± 0.8	$\mathbf{81.6} \pm 0.9$
Textures	69.0 ± 0.9	70.8 ± 0.9	68.3 ± 0.8	63.8 ± 0.8	$76.0 \pm -$	76.2 ± 0.7	77.0 ± 0.8	$\mathbf{78.9} \pm 0.8$
QuickDraw	42.6 ± 1.2	59.2 ± 1.2	50.3 ± 1.0	57.5 ± 1.0	$57.3 \pm -$	64.6 ± 0.9	64.3 ± 1.0	$\mathbf{71.5} \pm 1.0$
Fungi	38.2 ± 1.0	41.5 ± 1.2	41.4 ± 1.1	31.8 ± 1.0	$45.4 \pm -$	46.8 ± 1.1	$\mathbf{47.3} \pm 1.2$	47.3 ± 1.2
VGG Flower	85.5 ± 0.7	86.0 ± 0.8	87.3 ± 0.6	80.1 ± 0.9	$89.6 \pm -$	91.3 ± 0.5	89.9 ± 0.7	$\mathbf{93.5} \pm 0.8$
Traffic Sign	66.8 ± 1.3	60.8 ± 1.3	51.8 ± 1.0	46.5 ± 1.1	$66.2 \pm -$	82.5 ± 0.9	65.4 ± 1.1	$\mathbf{86.9} \pm 1.0$
MSCOCO	34.9 ± 1.0	48.1 ± 1.1	48.0 ± 1.0	41.4 ± 1.0	$\mathbf{55.7} \pm -$	55.2 ± 1.0	54.2 ± 1.1	54.4 ± 1.1
MNIST	-	-	-	80.8 ± 0.8	-	94.0 ± 0.5	88.1 ± 0.6	$\mathbf{94.6} \pm 0.6$
CIFAR-10	-	-	-	65.4 ± 0.8	-	$\mathbf{79.4} \pm 0.8$	70.4 ± 0.8	76.9 ± 0.8
CIFAR-100	-	-	-	52.7 ± 1.1	-	$\mathbf{70.5} \pm 0.9$	62.0 ± 1.0	69.3 ± 0.9
Average	57.0	61.4	59.1	55.8	65.0	71.7	68.5	**74.7**
Average Rank	6.0	5.2	5.6	6.5	4.0	2.4	3.0	**1.3**

TABLE V
TEST ACCURACY COMPARISON BETWEEN PRE-TRAINED REPRESENTATIONS: STANDARD VERSUS ROTATION-AUGMENTED

	1-shot		5-shot	
	standard	rotation	standard	rotation
*mini*IMAGENET	62.0 ± 0.4	64.5 ± 0.4	79.6 ± 0.3	81.5 ± 0.3
mini-60	63.9 ± 0.7	67.7 ± 0.5	81.5 ± 0.5	84.1 ± 0.5
*tiered*IMAGENET	69.1 ± 0.5	69.5 ± 0.6	83.9 ± 0.3	84.3 ± 0.4
tiered-780	78.0 ± 0.6	78.2 ± 0.6	89.9 ± 0.4	90.1 ± 0.4
H-Aircraft	79.2 ± 0.5	84.8 ± 0.3	89.4 ± 0.3	92.9 ± 0.2

F. Ablations on Pre-Training

Given the significance of pre-training on final performance, we investigate how the rotation data augmentation and data configuration impact the performance of the pre-trained representation. We focus on the effects of dataset sizes and the number of classes present in the dataset.

Rotation-Augmented Pre-training: In Section IV-A, we proposed to increase both the size and the number of classes in a dataset via input rotation. By rotating the input images by the multiples of $90°$, we quadruple both the size and the number of classes in a dataset. In Table V, we compare the performance of standard pre-training against the rotation-augmented one, for multiple datasets. We use the inferred labels from MeLa for pre-training.

The results suggest that rotation-augmented pre-training consistently improves the quality of the learned representation. It achieves over 3% improvements in both *mini*IMAGENET and H-aircraft, while obtains about 0.5% in *tiered*IMAGENET. It is clear that rotation augmentation works the best with smaller datasets with fewer classes. As the dataset increases in size and diversity, the additional augmentation has less impact on the learned representation.

Effects of Class Count: We further evaluate the effects of increasing number of classes in a dataset while maintaining the dataset size fixed. For this, we compare the performance of *mini*IMAGENET and *tiered*IMAGENET with their respective variants mini-60 and tiered-780.

Table V suggests that given a fixed size dataset, having more classes improves the quality of the learned representation compared to having more samples per class. We hypothesize that classifying more classes lead to more discriminative and robust features, while standard ℓ_2 regularization applied during pre-training prevents overfitting despite having fewer samples per class.

Overall, the experiments suggest that pre-training is a highly scalable process where increasing either data diversity or dataset size will lead to more robust representation for FSL. In particular, the number of classes in the dataset appears to play a more significant role than the dataset size.

G. Ablations on the Clustering Algorithm

The crucial component of MeLa is Algorithm 2, which infers a notion of global labels and allows pre-training to be exploited in GFSL setting. We perform several ablation studies to better understand Algorithm 2.

The Effects of No-replacement Sampling: We study the effects of no-replacement sampling, since it affects both the quality of the similarity measure through g_θ^{sim} and the number of tasks available for inferring global clusters. The results are shown in Table VI.

In Table VI, clustering accuracy is computed by assigning the most frequent ground truth label in each cluster as the desired target. Percentage of tasks clustered refers to the tasks that map to k unique clusters by Algorithm 2. The clustered tasks satisfy both constraints imposed by local labels and are used for pre-training.

For both sampling processes, MeLa achieves comparable performances across all three datasets. This indicates the robustness of Algorithm 2 in inferring suitable labels for pre-training, even when task samples do not repeat across tasks. This shows that Algorithm 2 is not trivially matching identical samples across task, but relying on g_θ^{sim} for estimating sample similarity. We note that mini-60 is particularly challenging under no-replacement sampling, with only 384 tasks in the meta-training set over 640 ground truth classes.

Effects of Pruning Threshold: In Algorithm 2, the pruning threshold is controlled by the hyper-parameter q. We investigate how different q values affect the number of clusters estimated

TABLE VI
THE EFFECTS OF NO-REPLACEMENT SAMPLING ON THE CLUSTERING ALGORITHM

Dataset	*mini*IMAGENET		*tiered*IMAGENET		mini-60	
Replacement	Yes	No	Yes	No	Yes	No
Tasks Clustered (%)	100	98.6	99.9	89.5	98.7	97.8
Clustering Acc (%)	100	99.5	96.4	96.4	72.8	70.1
1-shot Acc (%)	65.8 ± 0.4	65.8 ± 0.5	70.5 ± 0.5	70.5 ± 0.5	68.4 ± 0.7	68.4 ± 0.5
5-shot Acc (%)	82.8 ± 0.3	82.8 ± 0.4	85.9 ± 0.3	85.9 ± 0.3	84.0 ± 0.5	84.0 ± 0.5

TABLE VII
TEST ACCURACY (PRE-TRAIN ONLY) AND CLUSTER COUNT FOR VARIOUS PRUNING THRESHOLDS, 5-SHOT SETTING

*mini*IMAGENET (64 classes) Oracle Pre-train: 81.5%			*tiered*IMAGENET (351 classes) Oracle Pre-train: 84.5%			mini-60 (640 classes) Oracle Pre-train: 85.2%		
q	No. Clusters	MeLa	q	No. Clusters	MeLa	q	No. Clusters	MeLa
3	64	81.5 ± 0.4	3.5	351	84.3 ± 0.4	4.5	463	84.0 ± 0.4
4	75	81.1 ± 0.4	4.5	373	84.1 ± 0.3	5.5	462	83.8 ± 0.4
5	93	80.9 ± 0.4	5.5	444	84.0 ± 0.5	6.5	472	84.0 ± 0.4

TABLE VIII
TEST ACCURACY (PRE-TRAIN ONLY) USING ALGORITHM 2 VERSUS K-MEAN CLUSTERING

	*mini*IMAGENET			*tiered*IMAGENET		
Cluster Alg.	Cluster Acc	1-shot	5-shot	Cluster Acc	1-shot	5-shot
Alg. 2 (MeLa)	100	64.5 ± 0.4	81.5 ± 0.3	96.4	69.5 ± 0.5	84.3 ± 0.3
K-mean	84.9	60.7 ± 0.5	76.9 ± 0.3	28.2	64.8 ± 0.6	78.8 ± 0.5

by the algorithm and the corresponding test accuracy. Table VII suggest that MeLa is robust to a wide range of q and obtains representations similar to that produced by the ground truth labels. While it is possible to replace q with directly guessing the number of clusters in Algorithm 2, we note that tuning for q is more convenient since appropriate q values appear to empirically concentrate within a much narrower range, compared to the possible numbers of global clusters present in a dataset.

Inferred Labels versus Oracle Labels: From Tables VI and VII, we observe that it may be unnecessary to fully recover the oracle labels (when they exists). For mini-60, MeLa inferred 463 clusters over 640 classes, which implies mixing of the oracle classes. However, the inferred labels still perform competitively against the oracle labels, suggesting the robustness of the proposed method. The results also suggest that we may improve the recovery of the oracle labels by sampling more tasks from the meta-distribution.

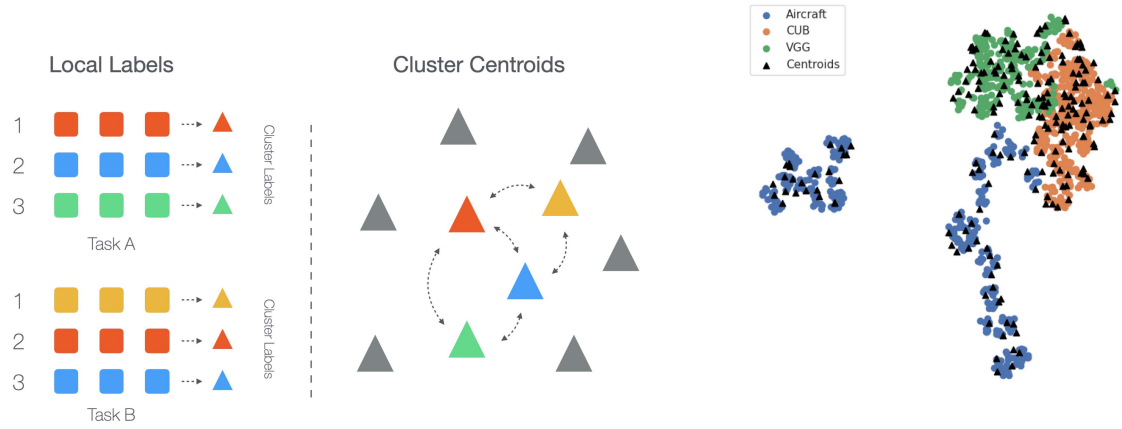
The Importance of Local Constraints: Algorithm 2 enforces consistent assignment of task samples given their local labels. To understand the importance of enforcing these constraints, we consider an ablation study where Algorithm 2 is replaced with the standard K-means algorithm. The latter is fully unsupervised and ignores any local constraints. We initialize the K-means algorithm with 64 clusters for *mini*IMAGENET and 351 clusters for *tiered*IMAGENET, matching the true class counts in respective datasets.

Table VIII indicates that enforcing local constraints is critical for generalization performance during meta-testing. In particular, test accuracy drops by over 5% for *tiered*IMAGENET, when the K-means algorithm ignores local task constraints. Among the two constraints, we note that (17) appears to be the more important one since nearly all tasks automatically match K unique clusters in our experiments (see tasks clustered in Table VI).

Domain Inference for multi-domain tasks: In addition to inferring global labels, We may further augment Algorithm 2 to infer the different domains present in a meta-training set, if we assume that all samples within a task belongs to a single domain. Given the assumption, two global clusters are connected if they both contain samples from the same task. This is illustrated in Fig. 2(a). Consequently, inferred clusters form an undirected graph with multiple connected components, with each representing a domain. We apply the above algorithm to the multi-domain Mixed Dataset consisting of Aircraft, CUB and Flower.

Fig. 2(b) visualizes the inferred domains on the multi-domain scenario. For each inferred cluster, we project its centroid onto a 2-dimensional point using UMAP [46]. Each connected component is assigned a different color. Despite some mis-clustering within each domain, we note that Algorithm 2 clearly separates the three domains present in the meta-training set and recovers them perfectly.

Domain inference is important for multi-domain scenario as it enables domain-specific pre-training. Recent works [e.g., [14], [37], [41] on Meta-Dataset have shown that combining domain-specific representation into a universal representation is empirically more advantageous than training on all domains together. Lastly, we remark that multi-domain meta-learning is also crucial for obtaining robust representation suitable for wider range of novel tasks, including cross-domain transfer.



(a) **Domain Inference via Connected Components**

(b) **UMAP visualization of different domains in multi-domain dataset**

Fig. 2. (a) The coloured clusters (red, green, blue and yellow) are connected since they both contains samples from the same task. Domains can be inferred by computing the connected components of the inferred clusters. (b) UMAP visualization of the three inferred domains from the 5-shot Mixed dataset containing Aircraft, CUB, and VGG. Circles are the means (using the pretrained features) of the instances in each task averaged per local class while triangles are the learned centroids, all vectors are embedded using UMAP. The three domains are recovered perfectly.

VI. Conclusion

In this work we focused on the role played by pre-training in meta-learning applications, with particular attention to few-shot learning problems. Our analysis was motivated by the recent popularity of pre-training as a key stage in most state-of-the-art FSL pipelines. We first investigated the benefits of pre-training from a theoretical perspective. We showed that in some settings this strategy enjoys significantly better sample complexity than pure meta-learning approaches, hence offering a justification for its empirical performance and wide adoption in practice.

We then proceeded to observe that pre-training requires access to global labels of the classes underlying the FSL problem. This might not always be possible, due to phenomena like heterogeneous labeling (i.e., multiple labelers having different labeling strategies) or contextual restrictions like privacy constraints. We proposed Meta-Label Learning (MeLa) as a strategy to address this concern. We compared MeLa with state-of-the-art methods on a number of tasks including well-established standard benchmarks as well as new datasets we designed to capture the above limitations on task labels. We observed that MeLa is always comparable or better than previous approaches and very robust to lack of global labels or the presence of conflicting labels.

More broadly, our work provides a solid foundation for understanding existing FSL methods, in particular the vital contribution of pre-training towards generalization performance. We also demonstrated that pre-training scales well with the size of datasets and data diversity, which in turn leads to more robust few-shot models. Future research may focus on further theoretical understanding of pre-training and better pre-training processes.

References

[1] A. Afrasiyabi, H. Larochelle, J.-F. Lalonde, and C. Gagné, "Matching feature sets for few-shot image classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 9014–9024.

[2] J.-B. Alayrac et al., "Flamingo: A visual language model for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 23716–23736.

[3] B. Amos and J. Z. Kolter, "OptNet: Differentiable optimization as a layer in neural networks," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 136–145.

[4] P. Bateni, R. Goyal, V. Masrani, F. Wood, and L. Sigal, "Improved few-shot visual classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 14493–14502.

[5] P. Bateni et al., "Beyond simple meta-learning: Multi-purpose models for multi-domain, active and continual few-shot learning," 2022, *arXiv:2201.05151*.

[6] L. Bertinetto, J. F. Henriques, P. H. Torr, and A. Vedaldi, "Meta-learning with differentiable closed-form solvers," in *Proc. Int. Conf. Learn. Representations*, 2019.

[7] T. Brown et al., "Language models are few-shot learners," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 1877–1901.

[8] W.-Y. Chen, Y.-C. Liu, Z. Kira, Y.-C. F. Wang, and J.-B. Huang, "A closer look at few-shot classification," in *Proc. Int. Conf. Learn. Representations*, 2018.

[9] Y. Chen, Z. Liu, H. Xu, T. Darrell, and X. Wang, "Meta-Baseline Exploring simple meta-learning for few-shot learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 9062–9071.

[10] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi, "Describing textures in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 3606–3613.

[11] G. Denevi, M. Pontil, and C. Ciliberto, "The advantage of conditional meta-learning for biased regularization and fine-tuning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 964–974.

[12] G. Denevi, M. Pontil, and C. Ciliberto, "Conditional meta-learning of linear representations," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 253–266.

[13] S. S. Du, W. Hu, S. M. Kakade, J. D. Lee, and Q. Lei, "Few-shot learning via learning the representation, provably," in *Proc. Int. Conf. Learn. Representations*, 2020.

[14] N. Dvornik, C. Schmid, and J. Mairal, "Selecting relevant features from a multi-domain representation for few-shot classification," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 769–786.

[15] A. El Baz et al., "Lessons learned from the neurips 2021 metadl challenge: Backbone fine-tuning without episodic meta-learning dominates for few-shot learning image classification," in *Proc. Int. Conf. Neural Inf. Process. Syst. Competitions Demonstrations Track*, 2022, pp. 80–96.

[16] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 594–611, Apr. 2006.

[17] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 1126–1135.

[18] L. Franceschi, P. Frasconi, S. Salzo, R. Grazzi, and M. Pontil, "Bilevel programming for hyperparameter optimization and meta-learning," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1568–1577.

[19] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "Dropblock: A regularization method for convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 10750–10760.

[20] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *Proc. Int. Conf. Learn. Representations*, 2018.

[21] M. Goldblum, S. Reich, L. Fowl, R. Ni, V. Cherepanova, and T. Goldstein, "Unraveling meta-learning: Understanding feature representations for few-shot tasks," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 3607–3616.

[22] N. Golowich, A. Rakhlin, and O. Shamir, "Size-independent sample complexity of neural networks," in *Proc. Conf. Learn. Theory*, 2018, pp. 297–299.

[23] I. Goodfellow et al., "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[24] D. Ha, A. Dai, and Q. V. Le, "Hypernetworks," 2016, *arXiv:1609.09106*.

[25] C. R. Harris et al., "Array programming with numpy," *Nature*, vol. 585, no. 7825, pp. 357–362, 2020.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[27] T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey, "Meta-learning in neural networks: A survey," 2020, *arXiv: 2004.05439*.

[28] S. Houben, J. Stallkamp, J. Salmen, M. Schlipsing, and C. Igel, "Detection of traffic signs in real-world images: The German traffic sign detection benchmark," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, 2013, pp. 1–8.

[29] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Comput. Sci. Eng.*, vol. 9, no. 3, pp. 90–95, 2007.

[30] G. Jerfel, E. Grant, T. Griffiths, and K. A. Heller, "Reconciling meta-learning and continual learning with online mixtures of tasks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 9119–9130.

[31] X. Jiang, M. Havaei, F. Varno, G. Chartrand, N. Chapados, and S. Matwin, "Learning to learn with conditional class dependencies," in *Proc. Int. Conf. Learn. Representations*, 2018.

[32] J. Jongejan, H. Rowley, T. Kawashima, J. Kim, and N. Fox-Gieg, "The quick, draw!-AI experiment," 2016. [Online]. Available: http://quickdraw.withgoogle.com,4

[33] A. Kolesnikov et al., "Big transfer (BiT): General visual representation learning," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 491–507.

[34] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 84–90.

[35] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum, "Human-level concept learning through probabilistic program induction," *Science*, vol. 350, no. 6266, pp. 1332–1338, 2015.

[36] K. Lee, S. Maji, A. Ravichandran, and S. Soatto, "Meta-learning with differentiable convex optimization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 10657–10665.

[37] W.-H. Li, X. Liu, and H. Bilen, "Universal representation learning from multiple domains for few-shot classification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 9526–9535.

[38] W.-H. Li, X. Liu, and H. Bilen, "Cross-domain few-shot learning with task-specific adapters," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 7161–7170.

[39] Z. Li, F. Zhou, F. Chen, and H. Li, "Meta-SGD: Learning to learn quickly for few-shot learning," 2017, *arXiv: 1707.09835*.

[40] T.-Y. Lin et al., "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[41] L. Liu, W. Hamilton, G. Long, J. Jiang, and H. Larochelle, "A universal representation transformer layer for few-shot image classification," 2020, *arXiv: 2006.11702*.

[42] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, Mar. 1982.

[43] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017, *arXiv: 1711.05101*.

[44] S. Maji, E. Rahtu, J. Kannala, M. B. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," 2013, *arXiv:1306.5151*.

[45] P. Mangla, N. Kumari, A. Sinha, M. Singh, B. Krishnamurthy, and V. N. Balasubramanian, "Charting the right manifold: Manifold mixup for few-shot learning," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis.*, 2020, pp. 2218–2227.

[46] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform manifold approximation and projection for dimension reduction," *arXiv: 1802.03426*.

[47] V. Mnih et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[48] A. Nichol, J. Achiam, and J. Schulman, "On first-order meta-learning algorithms," 2017, *arXiv: 1803.02999*.

[49] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Proc. IEEE 6th Indian Conf. Comput. Vis. Graph. Image Process.*, 2008, pp. 722–729.

[50] B. Oreshkin, P. R. López, and A. Lacoste, "TADAM: Task dependent adaptive metric for improved few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 721–731.

[51] A. Paszke et al., "Automatic differentiation in PyTorch," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017.

[52] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[53] H. Qi, M. Brown, and D. G. Lowe, "Low-shot learning with imprinted weights," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 5822–5830.

[54] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.

[55] A. Raghu, M. Raghu, S. Bengio, and O. Vinyals, "Rapid learning or feature reuse? Towards understanding the effectiveness of maml," 2019, *arXiv:1909.09157*.

[56] A. Ravichandran, R. Bhotika, and S. Soatto, "Few-shot learning with embedded class models and shot-free meta training," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 331–339.

[57] M. Ren et al., "Meta-learning for semi-supervised few-shot classification," in *Proc. Int. Conf. Learn. Representations*, 2018.

[58] J. Requeima, J. Gordon, J. Bronskill, S. Nowozin, and R. E. Turner, "Fast and flexible multi-task classification using conditional neural adaptive processes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 7959–7970.

[59] T. Ridnik, E. Ben-Baruch, A. Noy, and L. Zelnik-Manor, "ImageNet-21 k pretraining for the masses," 2021, *arXiv:2104.10972*.

[60] P. Rodríguez, I. Laradji, A. Drouin, and A. Lacoste, "Embedding propagation: Smoother manifold for few-shot classification," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 121–138.

[61] O. Russakovsky et al., "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[62] A. A. Rusu et al., "Meta-learning with latent embedding optimization," in *Proc. Int. Conf. Learn. Representations*, 2019.

[63] T. Saikia, T. Brox, and C. Schmid, "Optimized generic feature learning for few-shot classification across domains," 2020, *arXiv: 2001.07926*.

[64] B. Schroeder and Y. Cui, "FGVCx fungi classification challenge," 2018. [Online]. Available: https://github.com/visipedia/fgvcx_fungi_comp

[65] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*. New York, NY, USA: Cambridge Univ. Press, 2014.

[66] A. Shysheya, J. Bronskill, M. Patacchiola, S. Nowozin, and R. E. Turner, "FIT: Parameter efficient few-shot transfer learning for personalized and federated image classification," 2022, *arXiv:2206.08671*.

[67] D. Silver et al., "Mastering the game of Go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[68] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4077–4087.

[69] Q. Sun, Y. Liu, T.-S. Chua, and B. Schiele, "Meta-transfer learning for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 403–412.

[70] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, 2018, pp. 1199–1208.

[71] Y. Tian, Y. Wang, D. Krishnan, J. B. Tenenbaum, and P. Isola, "Rethinking few-shot image classification: A good embedding is all you need?," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 266–282.

[72] E. Triantafillou et al., "Meta-dataset: A dataset of datasets for learning to learn from few examples," in *Proc. Int. Conf. Learn. Representations*, 2019.

[73] E. Triantafillou, H. Larochelle, R. Zemel, and V. Dumoulin, "Learning a universal template for few-shot dataset generalization," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 10424–10433.

[74] J. Vanschoren, "Meta-learning," in *Proc. Automated Mach. Learn.*, 2019, pp. 35–61.

[75] O. Vinyals et al., "Matching networks for one shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3637–3645.

[76] R. Vuorio, S.-H. Sun, H. Hu, and J. J. Lim, "Multimodal model-agnostic meta-learning via task-aware modulation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 1–12.

[77] R. Wang, Y. Demiris, and C. Ciliberto, "Structured prediction for conditional meta-learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 2587–2598.
[78] R. Wang, M. Pontil, and C. Ciliberto, "The role of global labels in few-shot classification and how to infer them," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 27160–27170.
[79] P. Welinder et al., "Caltech-UCSD birds 200," Tech. Rep. California Institute of Technology, 2010.
[80] D. Wertheimer, L. Tang, and B. Hariharan, "Few-shot classification with feature map reconstruction networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 8012–8021.
[81] J. Xu and H. Le, "Generating representative samples for few-shot classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 9003–9013.
[82] S. Yang, L. Liu, and M. Xu, "Free lunch for few-shot learning: Distribution calibration," 2021, *arXiv:2101.06395*.
[83] H. Yao, Y. Wei, J. Huang, and Z. Li, "Hierarchically structured meta-learning," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7045–7054.
[84] H.-J. Ye, H. Hu, D.-C. Zhan, and F. Sha, "Few-shot learning via embedding adaptation with set-to-set functions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8808–8817.
[85] C. Zhang, Y. Cai, G. Lin, and C. Shen, "DeepEMD: Few-shot image classification with differentiable earth mover's distance and structured classifiers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 12200–12210.

Ruohan Wang received the PhD degree in machine learning from Imperial College under the supervision of prof. Yiannis Demiris and prof. Carlo Ciliberto, funded by Singapore National Science Scholarship. He is a research scientist with Institute for Infocomm Research, A*STAR Singapore. Previously, he was a postdoctoral Researcher with UCL's Intelligent Systems Group, supervised by Prof. Massimiliano Pontil. He has a broad interests in topics of machine learning, including representation learning, meta-learning, and imitation learning. His research goal is to design robust ML systems that could efficiently leverage past experiences and existing knowledge for future learning.

John Isak Texas Falk is currently working toward the PhD degree with the Department of Computer Science of UCL where he is supervised by Prof. Massimiliano Pontil and prof. Carlo Ciliberto. He is also a research fellow with the Italian Institute of Technology CSML under the supervision of prof. Massimiliano Pontil. His PhD is focused on few-shot and meta-learning in the context of kernel learning and statistical learning theory in the context of meta-learning, although he has an interest in representation learning, fairness and bandits. The goal of his research is how to design robust meta-learning systems that work well under real-world conditions and come with guarantees.

Massimiliano Pontil received the PhD degree in physics from the University of Genoa, in 1999. He is senior researcher with the Italian Institute of Technology (IIT), where he leads the Computational Statistics and Machine Learning group, and co-director of the ELLIS Unit Genoa, a joint effort of IIT and the University of Genoa. He is also part-time professor with University College London and member of the UCL Centre for Artificial Intelligence. He has made significant contributions to machine learning, particularly in the areas of kernel methods, multitask and transfer learning, sparsity regularization and statistical learning theory. He has published about 150 papers at international journals and conferences, is regularly on the programme committee of the main machine learning conferences, and has been on the editorial board of the Machine Learning Journal, Statistics and Computing, and JMLR. He has held visiting positions at a number of universities and research institutes, including the Massachusetts Institute of Technology, the Isaac Newton Institute for Mathematical Sciences in Cambridge, the City University of Hong Kong, the University Carlos III de Madrid, ENSAE Institute Polytechnique Paris, and Ecole Polytechnique.

Carlo Ciliberto received the bachelor's and master's degrees in mathematics from the Università Roma Tre (Magna Cum Laude), and the PhD degree in machine learning applied to robotics and computer vision from the Istituto Italiano di Tecnologia. He is associate professor with the Centre for Artificial Intelligence at University College London, He is member of the ELLIS society and of the ELLIS Unit based at UCL. He has been postdoctoral researcher with the Massachusetts Institute of Technology with the Center for Brain Minds and Machines and became lecturer (assistant professor) with Imperial College London before joining UCL, where he now carries out his main research activity. His research interests focus on foundational aspects of machine learning within the framework of statistical learning theory. He is particularly interested in the role of "structure" (being it in the form of prior knowledge or structural constraints) in reducing the sample complexity of learning algorithms with the goal of making them more sustainable both computationally and financially. He investigated these questions within the settings of structured prediction, multi-task and meta-learning, with applications to computer vision, robotics, and recommendation systems.

Open Access provided by 'Istituto Italiano di Tecnologia' within the CRUI CARE Agreement