Task Cooperation for Semi-Supervised Few-Shot Learning*

Han-Jia Ye, Xin-Chun Li, De-Chuan Zhan
State Key Laboratory for Novel Software Technology, Nanjing University
{yehj, lixc, zhandc}@lamda.nju.edu.cn

Abstract

Training a model with limited data is an essential task for machine learning and visual recognition. Few-shot learning approaches meta-learn a task-level inductive bias from SEEN class few-shot tasks, and the meta-model is expected to facilitate the few-shot learning with UNSEEN classes. Inspired by the idea that unlabeled data can be utilized to *smooth* the model space in traditional semi-supervised learning, we propose TAsk COoperation (TACO) which takes advantage of *unsupervised tasks* to *smooth* the *meta-model* space. Specifically, we couple the labeled support set in a few-shot task with easily-collected unlabeled instances, prediction agreement on which encodes the relationship between tasks. The learned *smooth* meta-model promotes the generalization ability on supervised UNSEEN few-shot tasks. The state-of-the-art few-shot classification results on *Mini*ImageNet and *Tiered*ImageNet verify the superiority of TACO to leverage unlabeled data and task relationship in meta-learning.

Introduction

Both instance collection and labeling costs influence the practical utility of a model in real-world applications, which requires a classifier to be trained with limited examples. For example, a robotic agent should be able to imitate behaviors from one single demonstration (Yu et al. 2018).

One solution to the Few-Shot Learning (FSL) problem takes advantage of data from related classes. Towards training effective classifiers for few-shot tasks with UNSEEN classes (a.k.a. the "meta-test" phase), meta-learning mimics the few-shot task evaluations on the SEEN class set (a.k.a. the "meta-train" set) and extracts task-level inductive bias in the "meta-training" phase (Baxter 2000; Vilalta and Drissi 2002; Maurer, Pontil, and Romera-Paredes 2016). For example, the instance embedding function (Vinyals et al. 2016; Snell, Swersky, and Zemel 2017), model initialization (Finn, Abbeel, and Levine 2017; Nichol, Achiam, and Schulman 2018), functional mapping (Qiao et al. 2018), and optimization strategies (Ravi and Larochelle 2017) facilitate FSL.

During meta-training, episodes of few-shot tasks, couples of few-shot support set and the same-distribution query set, are sampled from the SEEN class set to update the meta-model (as in Fig. 2 (a)). Specifically, a task-specific classifier is derived from the meta-model based on the few-shot support set, and the classifier's performance is measured on the corresponding query set. The supervision of meta-learning comes from the labels in the query set, so we define such tasks as the "supervised" ones. Similarly, we introduce *"unsupervised" task* as a task with a few-shot labeled support set and an *unlabeled* "pool" set. The pool set contains easily collected instances from any (even distractor) class, but it is difficult to provide supervision in meta-training directly.

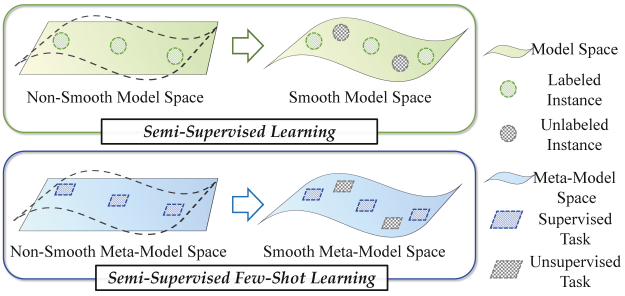


Figure 1: Analogy between semi-supervised learning (top) and *semi-supervised few-shot learning* (bottom), where unlabeled instances (resp. unsupervised tasks) assist shaping a *smooth* model (resp. meta-model) space. Limited labeled instances (resp. tasks) make revealing the characteristic of the data difficult (left). *Unsupervised tasks* facilitate a meta-model to generalize better and to construct close classifiers for similar tasks (right).

In this paper, we propose the TAsk COoperation (TACO) approach for few-shot classification, which takes advantage of the task relationship by incorporating both the supervised and *unsupervised tasks* during meta-training (we denote it as *Semi-Supervised Few-Shot Learning* (SS-FSL) in Fig. 2 (c)). As shown in Fig. 1 (bottom), directly learning the meta-model over supervised tasks could lead to a biased meta-model space, which constructs diverse classifiers for similar tasks. TACO makes the meta-model space *smooth*, so that similar support sets are mapped to close classifiers and the meta-model generalizes better. In detail, the similarity among few-shot tasks is measured by their prediction agree-

*De-Chuan Zhan is the corresponding author. This work is supported by NSFC (61773198, 6163000043, 61921006, 62006112), NSF of Jiangsu Province (BK20200313).

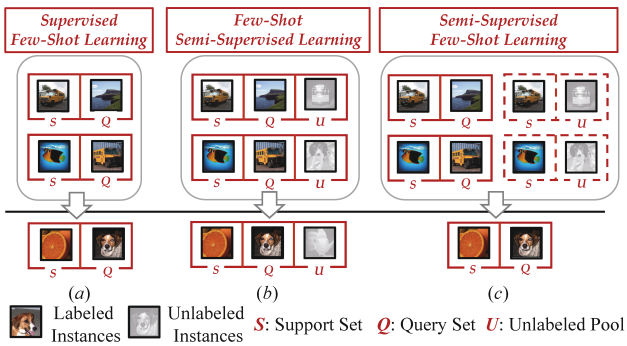


Figure 2: The difference between supervised few-shot learning (FSL), Few-Shot Semi-Supervised Learning (FS-SSL) and *Semi-Supervised Few-Shot Learning* (SS-FSL). In FSL (a), episodes of supervised tasks are sampled to train the meta-model (top), and it is the same scenario in the meta-test phase (bottom); In FS-SSL, each SEEN few-shot task is paired with an additional same-distributed unlabeled set (U) to enhance its ability *individually*, and correspondingly, the meta-model only learns how to utilize unlabeled data in a specific semi-supervised task; SS-FSL emphasizes taking advantage of the unlabeled data to construct *unsupervised tasks* (a joint set with S and U) from a *macro-perspective*, obtaining a *smooth* meta-model.

ment over the unlabeled pool set, which corresponds to the notion that *similar samples (resp. tasks) have similar labels (resp. classification behavior) in semi-supervised learning paradigm.* It is notable that unlabeled data are only used during meta-training to measure the smoothness, and the meta-model acts in a *fully supervised* manner in meta-test.

Several relatedness measures between tasks and few-shot classifiers are proposed and investigated for TACO. The same meta-learning mechanism extends various supervised few-shot approaches like ProtoNet (Snell, Swersky, and Zemel 2017) and ProtoMAML (Triantafillou et al. 2020). TACO variants achieve not only superior performance in different semi-supervised configurations but also get higher accuracy on fully supervised benchmarks like *Mini*ImageNet.

In summary, from the standpoint of traditional semi-supervised learning, we utilize unlabeled data from a *macro-perspective* to form *unsupervised few-shot tasks* and encourage close tasks to behave similarly. We propose TACO to incorporate the relationship between tasks to obtain a *smooth* meta-model space, and it can still generalize well even in fully supervised meta-test phase (without unlabeled data). The experimental results on both SS-FSL and standard FSL verify the effectiveness of the TACO approach.

Related Work

Training a model with limited examples is essential due to the instance collection and labeling costs (Li, Fergus, and Perona 2006; Lake et al. 2011; Lake, Salakhutdinov, and Tenenbaum 2015). Towards figuring out the data scarcity problem, two important paradigms, semi-supervised learning and meta-learning, are usually considered.

Semi-Supervised Learning (SSL) discovers the latent structure of data via unlabeled instances (Bennett and Demiriz 1998; Chapelle, Schlkopf, and Zien 2010; Oliver et al. 2018). To ensure *smoothness* of predictions, prediction consistency (Sajjadi, Javanmardi, and Tasdizen 2016), low entropy region (Grandvalet and Bengio 2004), and data generation (Kingma et al. 2014) act as key principles.

Meta-learning deals with the FSL problem by extracting task-level inductive bias from SEEN classes, and then generalizes to UNSEEN class few-shot tasks (Maurer, Pontil, and Romera-Paredes 2016; Chao et al. 2020). For example, the embedding-based (Vinyals et al. 2016; Snell, Swersky, and Zemel 2017; Lee et al. 2019), gradient-based (Finn, Abbeel, and Levine 2017; Nagabandi et al. 2019), and generative (Zhang et al. 2019) meta-learning methods.

Recent literature explores the usage of the unlabeled data in FSL. Transductive few-shot learning assumes all test instances come simultaneously, which are used as the unlabeled pool, so as to leverage the latent structure between training and test instances (Liu et al. 2019; Qiao et al. 2019). In Few-Shot Semi-Supervised Learning (FS-SSL), each task is equipped with an auxiliary set of unlabeled instances (even from distractor classes) in both meta-training and meta-test stages (Boney and Ilin 2017; Ayyad et al. 2019), and the meta-model learns to provide better classifier estimation based on the unlabeled data (Ren et al. 2018; Khodadadeh, Bölöni, and Shah 2019) (as in Fig. 2 (b)). Instead of formulating an SSL problem in each few-shot task, we focus on the *Semi-Supervised Few-Shot Learning* (SS-FSL) mechanism from a *macro-perspective*, where not only supervised tasks but also *unsupervised tasks* are incorporated during the meta-training (as in Fig. 2 (c)). Compared with FS-SSL, there are two main differences in our SS-FSL. First, the usage of unlabeled data is different. Different from forming episodes of semi-supervised tasks in FS-SSL, SS-FSL constructs *unsupervised tasks* to *smooth* the meta-model space from a *macro-perspective* — FS-SSL utilizes the unlabeled data to improve the ability of a specific few-shot task, while FS-SSL emphasizes improving the discriminative ability of the meta-model (*e.g.*, embeddings) with the help of *unsupervised tasks*. Second, meta-test strategies are different. Usually, FS-SSL needs the assistance of unlabeled data during meta-test, while SS-FSL can still generalize well even without unlabeled data owing to the *smooth* meta-model space.

Meta-Learning for Few-Shot Learning

In this section, we introduce the few-shot classification problem and describe how to solve it with meta-learning.

The Few-Shot Learning Problem

Few-Shot Learning (FSL) formalizes a classification task in the N-way K-shot form. The support set of a task $\mathcal{D}_{\mathbf{S}} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{NK}$ contains N classes and K labeled examples in each class, where the instance $\mathbf{x}_i \in \mathbb{R}^D$ and the one-hot coding label $\mathbf{y}_i \in \{0,1\}^N$. The goal of FSL is to train an N-way classifier $h \in \mathcal{H}_N : \mathbb{R}^D \to \{0,1\}^N$ based on the NK examples, where $\mathcal{H}_N$ is the N-way classifier space. h is

prone to over-fit when K is small (*e.g.*, $K = 1$). $\mathbf{sfx}(\mathbf{p})$ normalizes a vector $\mathbf{p} \in \mathbb{R}^N$ into a probability distribution with softmax, *i.e.*, $\sum_{n=1}^{N} \mathbf{sfx}(\mathbf{p})_n = 1$ and $\{\mathbf{sfx}(\mathbf{p})_n \geq 0\}_{n=1}^{N}$. Denote $\mathbf{KL}_U(\mathbf{p}\|\mathbf{q}) = \sum_{n=1}^{N} \mathbf{sfx}(\mathbf{p})_n \log \frac{\mathbf{sfx}(\mathbf{p})_n}{\mathbf{sfx}(\mathbf{q})_n}$ as an operator which normalizes two N-dimensional vectors with softmax and then outputs their KL-divergence.

Meta-Learning for Few-Shot Learning

Meta-learning learns a task-level mapping f from the N-way K-shot support set $\mathcal{D}_\mathbf{S}$ to its target classifier $h^* \in \mathcal{H}_N$ in a supervised way (Chao et al. 2020). To learn the meta-model f, episodes of tasks are sampled from a "meta-train" set with SEEN classes. In detail, each task contains a N-way K-shot support set $\mathcal{D}_\mathbf{S}$ and a query set $\mathcal{D}_\mathbf{Q}$ with same-distribution examples from the N classes. The quality of a meta-generated classifier $f(\mathcal{D}_\mathbf{S})$ is measured by its classification ability on $\mathcal{D}_\mathbf{Q}$. In summary, f can be learned by:

$$\min_{f} \sum_{(\mathcal{D}_\mathbf{S}, \mathcal{D}_\mathbf{Q})} \sum_{(\mathbf{x}_j^\mathcal{Q}, \mathbf{y}_j^\mathcal{Q}) \in \mathcal{D}_\mathbf{Q}} \ell\left(f(\mathcal{D}_\mathbf{S})(\mathbf{x}_j^\mathcal{Q}),\ \mathbf{y}_j^\mathcal{Q}\right) . \quad (1)$$

The summation of $(\mathcal{D}_\mathbf{S}, \mathcal{D}_\mathbf{Q})$ in Eq. 1 denotes the enumeration of all sampled tasks from the SEEN class set. The loss $\ell(\cdot, \cdot)$ measures the quality of a meta-generated classifier $f(\mathcal{D}_\mathbf{S})$ via the discrepancy between the predicted label and the ground-truth of the query set, *e.g.*, the cross-entropy. The lower the average loss when predicting instances in $\mathcal{D}_\mathbf{Q}$, the closer the meta-generated classifier to the target one. After optimizing Eq. 1, f maps a training set to its target classifier even with a few labeled examples. Since the meta-training mimics the few-shot evaluation, it is supposed to generalize to N-way K-shot tasks composed by UNSEEN classes (a.k.a. meta-test phase). The meta-model f could be implemented in a non-parametric style. In other words, a query instance $\mathbf{x}_j^\mathcal{Q}$ is classified based on a soft nearest neighbor rule:

$$\hat{y}_j = f(\mathcal{D}_\mathbf{S})(\mathbf{x}_j^\mathcal{Q}) = \sum_{(\mathbf{x}_i^\mathcal{S}, \mathbf{y}_i^\mathcal{S}) \in \mathcal{D}_\mathbf{S}} \mathbf{sim}\left(\phi(\mathbf{x}_j^\mathcal{Q}),\ \phi(\mathbf{x}_i^\mathcal{S})\right) \mathbf{y}_i^\mathcal{S}. \quad (2)$$

$\phi : \mathbb{R}^D \to \mathbb{R}^d$ extracts features of the input examples and transforms them into a latent space with d dimensions. $\mathbf{sim}(\phi(\mathbf{x}_j^\mathcal{Q}), \phi(\mathbf{x}_i^\mathcal{S}))$ measures the similarity between the query instance $\phi(\mathbf{x}_j^\mathcal{Q})$ and a support instance $\phi(\mathbf{x}_i^\mathcal{S})$.

Matching Network (Vinyals et al. 2016) uses the ℓ_2-normalized cosine similarity in Eq. 2. After learning ϕ with Eq. 1, the embedding facilitates the construction of nearest neighbor classifier. The Prototypical Network (Snell, Swersky, and Zemel 2017) implements Eq. 2 with the negative euclidean distance. When $K > 1$, it averages the same-class instances together and uses class centers (prototypes) for prediction. The embedding center of class n can be defined as $\mathbf{c}_n = \frac{1}{K} \sum_{y_{i,n}=1} \phi(\mathbf{x}_i^\mathcal{S})$, then we have $\hat{y}_j = \sum_{n=1}^{N} \mathbf{sim}(\phi(\mathbf{x}_j^\mathcal{Q}), \mathbf{c}_n)\mathbf{y}_n$.

Semi-Supervised Few-Shot Learning (SS-FSL). Considering the practical utility of unlabeled data, SS-FSL handles the case that most of the SEEN class data are unlabeled. The meta-model is required to utilize both labeled and unlabeled data during meta-training, while only labeled few-shot support set from UNSEEN classes are provided in meta-test.

Task Cooperation for Few-Shot Learning

We focus on the *Semi-Supervised Few-Shot Learning* (SS-FSL), using the unlabeled meta-train data to improve the generalization ability of the meta-model f. We first outline the main idea of TAsk COoperation (TACO) and then describe the concrete configurations. Last are discussions.

TACO for Semi-Supervised Few-Shot Learning

Towards incorporating the easily collected and informative unlabeled data during the meta-training, we propose the TAsk COoperation (TACO) framework where the related tasks cooperate with each other for a *smooth* meta-model f. Traditional semi-supervised learning assumes that a *smooth* function maps near inputs to similar outputs, which is an essential property to achieve discriminative and generalizable models (Friedman, Hastie, and Tibshirani 2001; Chapelle, Schlkopf, and Zien 2010; Berthelot et al. 2019).

For an N-way K-shot classification task, f maps its support set $\mathcal{D}_\mathbf{S}$ to its corresponding classifier $h_N = f(\mathcal{D}_\mathbf{S})$. TACO generalizes the *smooth* notion in the traditional supervised learning to the meta-model space. From a *macro-perspective* of meta-learning, we first make an *analogy* between the training instance in the traditional supervised learning and the few-shot support set in the meta-learning.

We propose TACO to better capture the task relationship, which adds a *smoothness* constraint over the meta-learning objective in Eq. 1, so that two close tasks behave similarly:

$$\lambda \sum_{(\mathcal{D}_\mathbf{S}, \hat{\mathcal{D}}_\mathbf{S})} \mathbf{DIS}\left(f(\mathcal{D}_\mathbf{S}),\ f(\hat{\mathcal{D}}_\mathbf{S})\right) . \quad (3)$$

We assume $\mathcal{D}_\mathbf{S}$ and $\hat{\mathcal{D}}_\mathbf{S}$ are two visually/semantically similar few-shot support sets sampled from the meta-train set, and $\mathbf{DIS}(\cdot, \cdot)$ measures the discrepancy between two mapped models in $\mathcal{H}_N$. $\lambda > 0$ is a balance parameter. By minimizing Eq. 3 together with Eq. 1, the meta-model f not only maps a task to its target classifier but also generates similar classifiers for close few-shot support set (revealed by the small classifier-space distance between $f(\mathcal{D}_\mathbf{S})$ and $f(\hat{\mathcal{D}}_\mathbf{S})$), which corresponds well to the *smooth* notion in traditional supervised/semi-supervised learning. Benefited from TACO, the meta-model f becomes *smooth* and more discriminative, generalizing better in the meta-test stage.

Similarity Measures for TACO

Eq. 3 leaves the question of how to define the similarity between few-shot support sets and the discrepancy between classifiers. Here we provide detailed definitions.

Similarity between tasks. Given an N-way K-shot support set $\mathcal{D}_\mathbf{S}$, we generate another "similar" few-shot support set $\hat{\mathcal{D}}_\mathbf{S}$ based on two strategies. First, we consider two tasks are similar if they have the same set of classes. Given the N classes in $\mathcal{D}_\mathbf{S}$, we sample another non-overlapping

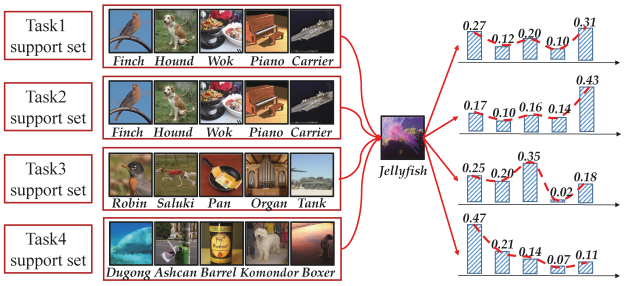


Figure 3: Empirical observations of measuring similarity between classifiers based on their predictions over the instance space. Each row corresponds to a 5-way task trained by 3000 examples in *Mini*ImageNet, where the task-specific classifier is based on Nearest Center Mean (NCM) over embeddings. The first two tasks have exactly the same categories while their classifiers are learned with different initializations; the third task has different classes but all classes belong to the same super-categories with the first two; the last task targets classes from different sets of super-classes. The construction of tasks encodes the task-level visual and semantic similarity. The similarity between the output of meta-model, *i.e.*, the classifiers, could be revealed by their prediction results over the same (even distractor class) instance, as shown in the r.h.s. The histograms demonstrate the mean prediction results of 600 examples from the "Jellyfish" class. The visually/semantically similar tasks will have closer NCM decisions compared with the dissimilar ones.

K instances from the labeled meta-train set for each of the N classes to construct $\hat{\mathcal{D}}_{\mathbf{S}}$. Besides, we keep the order of classes in the two few-shot classification tasks the same, which maintains a correspondence between their classifiers. Second, we borrow the idea from standard semi-supervised learning (Sajjadi, Javanmardi, and Tasdizen 2016; Berthelot et al. 2019) to construct similar tasks based on perturbations. In this case, instances in $\hat{\mathcal{D}}_{\mathbf{S}}$ are the same as those instances in $\mathcal{D}_{\mathbf{S}}$ except additional (advanced) data augmentation operations (*e.g.*, random crop). Data augmentation changes the raw image input of an instance to some extent while keeping its semantic meaning, so the transformed task is close to the original one. Two labeled tasks with similar support sets usually target similar classifiers, so it is meaningful to apply the task similarity objective to them.

Remark 1 *In addition to generating visually similar tasks, we can also obtain semantically similar tasks based on class relationships, e.g., the binary classification task for "tiger vs. dog" and "cat vs. dog" are similar. Since such a class-wise similarity measure requires semantic attributes for classes, we leave it for future study.*

Similarity between Classifiers. The output of the meta-model f w.r.t. a support set, an N-way classifier h_N, is a function from instances to labels, and directly measuring the discrepancy between two classifiers in Eq. 3 requires function space metrics. Since the characteristic of classifiers could be revealed by its predictions $\{h_N(\mathbf{x})\}$ over all

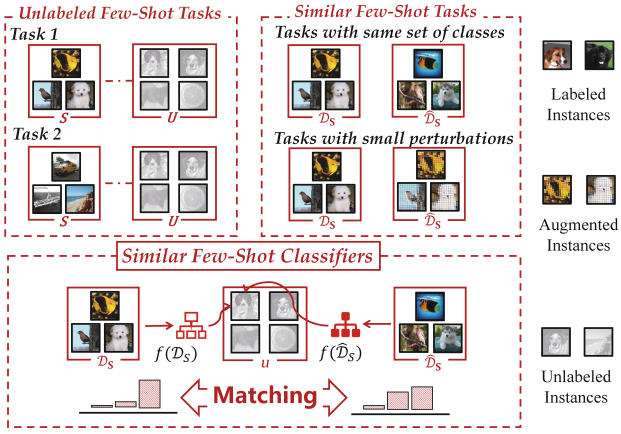


Figure 4: Illustration on the usage of *unsupervised tasks* for SS-FSL in TACO. Similarities for task and classifier are proposed to ensure the *smoothness* of meta-model f.

possible instances $\{\mathbf{x}\}$ sampled from the task distribution, we transform the similarity between two classifiers *from the function space to the instance space* — two similar classifiers have similar predictions over all instances.

We first empirically demonstrate that the prediction for an object from other classes except the ones in few-shot support set (*i.e.*, distractor classes) still reveals the properties of the embedding based few-shot classifier. Consider a task discerning N cat classes, the predictions of its classifier on a dog image decomposes the cat-level characteristic into these N cat classes. With classifiers based on embeddings, the confidence of a new instance implies its similarity to those N class centers in a joint embedding space. We verify this point with Nearest Center Mean (NCM) Classifier (Mensink et al. 2013) based on the tasks from *Mini*ImageNet (in Fig. 3). The agreements between normalized confidences demonstrate the similarity among tasks (and their target classifiers) — visually/semantically similar few-shot support sets make similar predictions on instances.

Based on the previous observation, we take a further step to make use of the unlabeled data during meta-training to measure the similarity between classifiers. Since the query set error $\sum_{(\mathbf{x}_j^{\mathcal{Q}},\mathbf{y}_j^{\mathcal{Q}})\in\mathcal{D}_{\mathbf{Q}}} \ell\left(h_N(\mathbf{x}_j^{\mathcal{Q}}),\ \mathbf{y}_j^{\mathcal{Q}}\right)$ is used to update the meta-model f, it is notable that if we can not get access to $\mathbf{y}_j^{\mathcal{Q}}$, *i.e.*, the query set labels, the meta-model f can not be updated. Thus, we make another analogy between supervised "examples" (*i.e.*, the instance and label pair) in the traditional supervised learning and the tasks (*i.e.*, the "support" and "query" sets pair) in the meta-learning. The "instance" and "label" in the supervised learning correspond to the "few-shot support set" and "query set" in the meta-learning, respectively. A few-shot support set with an unlabeled query set is similar to the unlabeled instances in traditional supervised learning. We denote a supervised task as a couple of labeled support and query sets $(\mathcal{D}_{\mathbf{S}}, \mathcal{D}_{\mathbf{Q}})$, and an *unsupervised task* as a combination of a few-shot labeled support set $\mathcal{D}_{\mathbf{S}}$ and an unlabeled pool set $\mathcal{D}_{\mathbf{pool}} = \{\mathbf{x}_k^{\mathcal{P}}\}$ sampled from the unlabeled part of the meta-train set. In-

stances in $\mathcal{D}_{\mathbf{pool}}$ may come from distractor classes w.r.t. those classes in $\mathcal{D}_{\mathbf{S}}$. During SS-FSL, for two similar *unsupervised tasks* $(\mathcal{D}_{\mathbf{S}}, \mathcal{D}_{\mathbf{pool}})$ and $(\hat{\mathcal{D}}_{\mathbf{S}}, \mathcal{D}_{\mathbf{pool}})$ sharing the same unlabeled pool set $\mathcal{D}_{\mathbf{pool}}$, we measure the similarity between the output of the meta-model — the similarity between two few-shot classifiers — based on the JS divergence of their predicted distributions on $\mathcal{D}_{\mathbf{pool}}$:

$$\begin{aligned}&\mathbf{DIS}(f(\mathcal{D}_{\mathbf{S}}),\ f(\hat{\mathcal{D}}_{\mathbf{S}})) \qquad (4)\\ &= \sum_{\mathbf{x}_k^{\mathcal{P}} \in \mathcal{D}_{\mathbf{pool}}} \mathbf{JSD}_T\left(f(\mathcal{D}_{\mathbf{S}})(\mathbf{x}_k^{\mathcal{P}}) \parallel f(\hat{\mathcal{D}}_{\mathbf{S}})(\mathbf{x}_k^{\mathcal{P}})\right).\end{aligned}$$

$f(\mathcal{D}_{\mathbf{S}})(\mathbf{x}_k^{\mathcal{P}})$ provides the affiliation confidence of an instance $\mathbf{x}_k^{\mathcal{P}}$ towards the N classes in $\mathcal{D}_{\mathbf{S}}$. $\mathbf{JSD}_T(\mathbf{p}\|\mathbf{q}) = \frac{1}{2}\mathbf{KL}_U(\mathbf{p}\|\frac{\mathbf{q}}{T}) + \frac{1}{2}\mathbf{KL}_U(\mathbf{q}\|\frac{\mathbf{p}}{T})$ is the JS divergence over the unnormalized predictions, the smaller the value, the closer these two distributions. T is a positive temperature to soften the predicted distribution (Hinton, Vinyals, and Dean 2015; Ye, Lu, and Zhan 2020). In our experiments, we stop the gradient of the second vector when computing the JS divergence. By matching the predictions of two few-shot classifiers on the same set of instances without label, the meta-model is required to map similar few-shot tasks to similar-performed models. Therefore, the outputs of similar meta-model inputs are pulled together, which forces the *smoothness* of the meta-model f. Eq. 4 could be applied to two labeled tasks if we replace $\mathcal{D}_{\mathbf{pool}}$ as the union of query sets.

Remark 2 *Minimizing the discrepancy between similar tasks' predictions potentially produces a* smooth *instance embedding encoder ϕ. However, matching the embeddings directly from the input perspective of the meta-model is not as flexible as Eq. 4, which is too strong and does not work well in our experiments.*

Objective. TACO uses Eq. 4 as an auxiliary objective, *i.e.*,

$$\begin{aligned}\min_f &\sum_{(\mathcal{D}_{\mathbf{S}}, \mathcal{D}_{\mathbf{Q}}, \mathcal{D}_{\mathbf{pool}})} \sum_{(\mathbf{x}_j^{\mathcal{Q}}, \mathbf{y}_j^{\mathcal{Q}}) \in \mathcal{D}_{\mathbf{Q}}} \ell\left(f(\mathcal{D}_{\mathbf{S}})(\mathbf{x}_j^{\mathcal{Q}}),\ \mathbf{y}_j^{\mathcal{Q}}\right) \qquad (5)\\ &+ \lambda \sum_{\mathbf{x}_k^{\mathcal{P}} \in \mathcal{D}_{\mathbf{pool}} \bigcup \mathcal{D}_{\mathbf{Q}}} \mathbf{JSD}_T\left(f(\mathcal{D}_{\mathbf{S}})(\mathbf{x}_k^{\mathcal{P}}) \parallel f(\hat{\mathcal{D}}_{\mathbf{S}})(\mathbf{x}_k^{\mathcal{P}})\right).\end{aligned}$$

TACO can be instantiated with the embedding-based methods like ProtoNet. Eq. 5 acts as an efficient way to maximize the similarity between both labeled and *unsupervised* tasks. During meta-training, we sample $\mathcal{D}_{\mathbf{S}}$, $\mathcal{D}_{\mathbf{Q}}$ and $\mathcal{D}_{\mathbf{pool}}$ from labeled and unlabeled meta-train set respectively (as in Fig. 4 and Alg.1). To take full advantage of examples, we combine the query set $\mathcal{D}_{\mathbf{Q}}$ with $\mathcal{D}_{\mathbf{pool}}$. The prediction matching not only utilizes the unlabeled meta-train data in a semi-supervised manner, but also promotes the co-supervision between similar tasks. During meta-test with $\mathcal{D}_{\mathbf{S}}$ only, a discriminative classifier is generated based on f in a supervised manner *without additional unlabeled instances*.

Remark 3 TACO *is general based on the definition of the similarity measurements of inputs (few-shot support sets) and outputs (target classifiers) of the meta-model. Since the embedding based classifiers project all the instances into a common subspace, it is able to measure the cross-class similarity in an unsupervised way between in-task class instances and distractor class instances. In our experiments, we implement f with ProtoNet (Snell, Swersky, and Zemel 2017) and ProtoMAML (Triantafillou et al. 2020) (in the supplementary). By minimizing the discrepancy between similar classifiers,* TACO *updates the meta-model (i.e., the embedding) to pull similar few-shot tasks together, which gives rise to a* smooth *task-classifier meta-model (Saito et al. 2018). Thus, given the neighborhood* SEEN *class few-shot task w.r.t. an* UNSEEN *class few-shot task, the discerning ability of a well-performed meta-model on those similar* SEEN *class tasks generalize to the* UNSEEN *tasks as well.*

Algorithm 1 The meta-training flow of the TACO.

Require: SEEN class set $\mathcal{S}$
for all iteration = 1,... **do**
  Sample N-way K-shot $(\mathcal{D}_{\mathbf{S}}, \mathcal{D}_{\mathbf{Q}})$ from $\mathcal{S}$
  Generate similar tasks $\hat{\mathcal{D}}_{\mathbf{S}}$ based on $\mathcal{D}_{\mathbf{S}}$
  Sample $\mathcal{D}_{\mathbf{pool}}$ from the unlabeled part of $\mathcal{S}$
  for all $(\mathbf{x}_j^{\mathcal{Q}}, \mathbf{y}_j^{\mathcal{Q}}) \in \mathcal{D}_{\mathbf{Q}}$ **do**
    Get $f(\mathcal{D}_{\mathbf{S}})(\mathbf{x}_j^{\mathcal{Q}})$
  end for
  for all $\mathbf{x}_k^{\mathcal{P}} \in \mathcal{D}_{\mathbf{Q}} \bigcup \mathcal{D}_{\mathbf{pool}}$ **do**
    Get $\mathbf{JSD}_T(f(\mathcal{D}_{\mathbf{S}})(\mathbf{x}_k^{\mathcal{P}}) \parallel f(\hat{\mathcal{D}}_{\mathbf{S}})(\mathbf{x}_k^{\mathcal{P}}))$
  end for
  Compute objective as in Eq. 5 and update f
end for
return Few-shot classifier mapping f

Experiments

We investigate TACO on *Mini*ImageNet as well as *Tiered*ImageNet. We describe experimental setups, and then provide the few-shot classification performance together with visualization results. Implementation details, qualitative and quantitative evaluations are in the supplementary.

Experimental Setups

Datasets. *Mini*ImageNet (Vinyals et al. 2016) and *Tiered*ImageNet (Ren et al. 2018) contain 100 classes and 608 classes respectively. All images are resized to $3\times84\times84$ following (Vinyals et al. 2016; Finn, Abbeel, and Levine 2017; Snell, Swersky, and Zemel 2017). We use the standard split of two datasets following (Ravi and Larochelle 2017; Ren et al. 2018), where meta-train, meta-val, and meta-test have non-overlapping classes.

Supervised Evaluation Protocols. We evaluate mean accuracy over 10,000 5-way 1-Shot and 5-Way 5-shot tasks (Vinyals et al. 2016; Ye et al. 2020), where the test set in a task has 15 examples from each of the 5 classes. In this supervised evaluation, all labeled examples in the meta-train set are utilized. We omit the 95% confidence interval in the experiments, and detailed values are in the supplementary.

Semi-Supervised Data Generation and Evaluation Protocols. We construct the semi-supervised meta-train set by removing part of its labels. Two different partitions are investigated. The first strategy splits all examples in the meta-train set across "classes" (SAC). In this case, we randomly select 30% classes in the meta-train set as the labeled part and uses the instances in the remaining classes without their labels as the unlabeled set. Similarly, we randomly select 30% instances across "instances" (SAI). In the SAI case, it is possible to sample non-distractor classes from the unlabeled pool, which reduces the classification difficulty w.r.t. SAC to some extent. Based on the observation in (Oliver et al. 2018), it is more realistic to *set the number of images in the meta-val set smaller than the number of labeled instances in the meta-train set*. Thus instead of preserving the whole meta-val set, we adopt the same SAC or SAI split methods to *reduce the size of the meta-val set*. Only selected labeled meta-val images are utilized to select the best model. The average performance of 3 random partitions is reported.

Comparison Methods. We mainly compare TACO with four embedding based approaches, namely MatchNet (Vinyals et al. 2016), ProtoNet (Snell, Swersky, and Zemel 2017), Semi-ProtoNet (Ren et al. 2018) and PRWN (Ayyad et al. 2019). Semi-ProtoNet is designed for few-shot semi-supervised learning, and we adapt it in our setting. Its improved version, with an MLP-based selector to detect helpful unlabeled instances, is denoted as Semi-ProtoNet*. PRWN gets compact and well-separated class representations via prototypical random walk.

Implementation Details. We use a 4-layer ConvNet (Vinyals et al. 2016; Finn, Abbeel, and Levine 2017; Snell, Swersky, and Zemel 2017) as the backbone, which is initialized following (Rusu et al. 2018; Ye et al. 2020). TACO and all comparison methods are fine-tuned on the pre-trained embedding in meta-training, and For semi-supervised FSL, we sample 75 unlabeled instances in each mini-batch. We also investigate the ResNet-12 (Lee et al. 2019), which is complicated but with high discriminative ability. We find ResNet over-fits when applied to the SS-FSL setting with a limited number of meta-train data, so we only test ResNet in the standard supervised few-shot classification tasks. Meta-model in TACO is implemented with ProtoNet. For constructing similar *unsupervised tasks*, we sample two support sets with the same classes and then apply random perturbations to them via the advanced augmentation reported in (Xie et al. 2020).

Results of Semi-Supervised Few-Shot Learning

We first investigate TACO for *Semi-Supervised Few-Shot Learning* (SS-FSL) case on two benchmark datasets *Mini*ImageNet and *Tiered*ImageNet. Results are recorded in Table 1 and Table 2. All compared methods are required to meta-learn few-shot facilitated embeddings whose qualities are revealed by the average accuracy on the meta-test set. Two split strategies, *i.e.*, split across classes and instances, are considered to verify the importance of the unlabeled meta-train set. Due to the fact there are more diverse examples (more classes) in the SAI case, all few-shot methods achieve better classification accuracy in this scenario compared with the SAC case.

*Mini*ImageNet	1-shot		5-shot	
Configuration→	SAC	SAI	SAC	SAI
MatchNet	42.33	44.73	55.93	59.03
ProtoNet	43.18	45.41	54.80	58.41
SemiProto	42.41	44.22	58.70	61.54
SemiProto*	42.84	45.59	59.21	62.31
PRWN	42.72	44.65	58.90	61.34
TACO	**43.97**	**46.56**	**61.13**	**62.85**

Table 1: Mean SS-FSL accuracy over 10,000 tasks on *Mini*ImageNet, with only 30% labeled meta-train set. SAC/SAI denote Split (meta-train set) Across Classes/Instances.

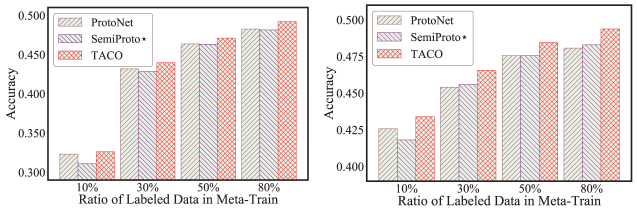
*Tiered*ImageNet	1-shot		5-shot	
Configuration→	SAC	SAI	SAC	SAI
MatchNet	49.72	52.12	64.62	67.17
ProtoNet	50.24	52.42	67.74	70.56
SemiProto	49.29	51.31	68.52	70.88
SemiProto*	50.41	51.78	**68.92**	71.17
PRWN	50.28	51.35	67.96	70.23
TACO	**51.82**	**54.66**	68.77	**71.83**

Table 2: SS-FSL accuracy on the *Tiered*ImageNet, with 30% labeled meta-train set.

MatchNet and ProtoNet are meta-trained in a fully supervised manner, where only the labeled meta-train set is used. Semi-Proto takes advantage of the unlabeled instances to help estimate the center of each class during meta-training. Since there are no unlabeled data in the meta-test phase, Semi-Proto does not improve a lot w.r.t. the vanilla ProtoNet in the 1-Shot scenario. From the results, the unlabeled instances help a lot when there is more than one instance in each class, and the distractor detector in Semi-Proto* works well especially in this case. Table 1 and Table 2 provide consistent results, where the methods taking advantage of the unlabeled data in meta-training achieve (at least slightly) better results than the supervised counterparts. The supervised baselines are really strong. The same phenomenon is also discovered in the classical deep semi-supervised learning (Oliver et al. 2018). TACO uses the unlabeled data by matching model predictions. With high-quality embeddings, TACO gets the best performance in both cases over the two benchmarks even there are no unlabeled instances during meta-test. It verifies TACO is able to meta-learn more discriminative meta-knowledge (instance embeddings) with the unlabeled data in meta-training.

T=2	SAC	SAI	$\lambda = 0.1$	SAC	SAI
$\lambda = 0$	42.33	44.73	$T = 1$	43.58	46.50
$\lambda = 0.1$	43.97	46.56	$T = 2$	43.97	46.56
$\lambda = 1$	42.31	44.65	$T = 4$	43.23	46.47

Table 3: Semi-supervised 1-shot classification accuracy on *Mini*ImageNet with ConvNet backbone, where only 30% of meta-train set are labeled. Performance of TACO using different parameters are compared.



(*a*) Split Across Classes (*b*) Split Across Instances

Figure 5: The change of 5-Way 1-Shot classification *accuracy* of ProtoNet, Semi-ProtoNet⋆, and TACO when the *ratio of labeled examples in the meta-train set* changes. Both label splits across classes and instances are investigated.

Ablation Studies

Influence of parameters. There are two main parameters in TACO, the balance weight λ weighting the distribution matching term, and the temperature T to soften the predicted confidence. We show the influences of these two parameters in the SS-FSL scenario. The same configurations are used as the previous experiments. From the results in Table 3, we find the distribution matching term indeed improves the learned embedding upon the supervised baseline ($\lambda = 0$). For the reason that we try to match the prediction distributions of two tasks mutually, the temperature used to scale the prediction outputs does not influence the performance a lot.

Influence of the Label Ratio Change. We also test the TACO approach when the ratio of labeled meta-train set varies, ranging from 10%, 30%, 50% to 80%. The remaining part of the labeled set in the meta-train set is used as the unlabeled set for SS-FSL. Results of 5-way 1-shot classification accuracy with both SAC and SAI partitions are shown in Fig. 5. Two plots reveal the same trends that with more labeled instances in the meta-training, all few-shot approaches achieve better performance. Among all results, TACO gets the best 5-way 1-shot classification results in all cases, especially meta-learned in SAI partition. Semi-Proto⋆ cannot improve the quality of the embedding especially when the size of the labeled data is very small (*e.g.*, 10%). The results verify the robustness of TACO.

Supervised Few-Shot Classification

As mentioned before and for fair comparisons, we investigate the supervised few-shot classification via replacing $\mathcal{D}_{\mathbf{pool}}$ by the union of query sets of two supervised few-shot tasks in Eq. 4. We find that TACO still works due to its explicit consideration of task relationship. The *smoothness* of a meta-learned model improves the generalization ability of the learned embedding when classifying UNSEEN few-shot tasks. We compare TACO with TEAM (Qiao et al. 2019), TPN (Liu et al. 2019), TapNet (Yoon, Seo, and Moon 2019), MTL (Sun et al. 2019), MetaOpt (Lee et al. 2019), CAN (Hou et al. 2019) on *Mini*ImageNet and *Tiered*ImageNet datasets with the ResNet backbone, the results are shown in Table 4. For *Mini*ImageNet, we cite the published results of compared methods, and we can find that TACO can get better performances, which can also be verified from *Tiered*ImageNet.

	*Mini*ImageNet		*Tiered*ImageNet	
5-Way	1-Shot	5-Shot	1-Shot	5-Shot
TapNet	61.65	76.36	63.08	80.26
MTL	61.20	75.50	65.60	78.60
MetaOpt	62.64	78.63	65.99	81.56
CAN	63.85	79.44	69.89	84.23
TACO (**Ours**)	**66.57**	**82.10**	**71.12**	**85.42**
TEAM†	60.07	75.9	-	-
TPN†	59.46	75.65	-	-
CAN†	67.19	80.64	73.21	84.93
TACO†(**Ours**)	**68.23**	**83.42**	**75.53**	**85.72**

Table 4: Supervised few-shot classification accuracy on the *Mini*ImageNet and *Tiered*ImageNet using the ResNet-12 Backbone. "†" denotes the transductive FSL method which utilizes the unlabeled data from the query set.

In addition to the fully supervised comparison, we also apply TACO in a transductive manner (super-scripted by "†" in Table 4), where the query set acts as the unlabeled pool. By taking advantage of unlabeled data in each few-shot task as the Semi-ProtoNet (Ren et al. 2018) manner, TACO promote the FSL performance especially in the 1-shot scenario. More details could be found in the supplementary.

Conclusion

Instead of utilizing unlabeled data to help classification in each few-shot task, we focus on the *Semi-Supervised Few-Shot Learning* (SS-FSL) problem from a *macro-perspective*. For a pair of meta-training tasks, the proposed TAsk COoperation (TACO) approach leverages *unsupervised tasks* — couples of a labeled few-shot support set and an unlabeled query set with distractor classes — to minimize the disagreement of predictions between their few-shot classifiers. Thus, TACO obtains a *smooth* meta-model space where similar few-shot tasks have close classifiers, which leads to a more discriminative and generalizable meta-model. Finally, a supervised classifier could be effectively constructed when targeting UNSEEN class few-shot tasks. TACO improves FSL performance on two benchmarks in both semi-supervised and supervised scenarios. Future work includes extending the TACO paradigm to a fully unsupervised scenario.

References

Ayyad, A.; Navab, N.; Elhoseiny, M.; and Albarqouni, S. 2019. Semi-Supervised Few-Shot Learning with Prototypical Random Walks. *CoRR* abs/1903.02164v3.

Baxter, J. 2000. A Model of Inductive Bias Learning. *JAIR* 12: 149–198.

Bennett, K. P.; and Demiriz, A. 1998. Semi-Supervised Support Vector Machines. In *NeurIPS*, 368–374.

Berthelot, D.; Carlini, N.; Goodfellow, I.; Papernot, N.; Oliver, A.; and Raffel, C. A. 2019. Mixmatch: A holistic approach to semi-supervised learning. In *NeurIPS*, 5049–5059.

Boney, R.; and Ilin, A. 2017. Semi-Supervised Few-Shot Learning with Prototypical Networks. *CoRR* abs/1711.10856.

Chao, W.-L.; Ye, H.-J.; Zhan, D.-C.; Campbell, M.; and Weinberger, K. Q. 2020. Revisiting Meta-Learning as Supervised Learning. *CoRR* abs/2002.00573.

Chapelle, O.; Schlkopf, B.; and Zien, A. 2010. *Semi-Supervised Learning*. The MIT Press.

Finn, C.; Abbeel, P.; and Levine, S. 2017. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In *ICML*, 1126–1135.

Friedman, J.; Hastie, T.; and Tibshirani, R. 2001. *The elements of statistical learning*, volume 1. Springer series in statistics New York.

Grandvalet, Y.; and Bengio, Y. 2004. Semi-supervised Learning by Entropy Minimization. In *NeurIPS*, 529–536.

Hinton, G. E.; Vinyals, O.; and Dean, J. 2015. Distilling the Knowledge in a Neural Network. *CoRR* abs/1503.02531.

Hou, R.; Chang, H.; Ma, B.; Shan, S.; and Chen, X. 2019. Cross Attention Network for Few-shot Classification. In *NeurIPS*, 4005–4016.

Khodadadeh, S.; Bölöni, L.; and Shah, M. 2019. Unsupervised Meta-Learning for Few-Shot Image Classification. In *NeurIPS*, 10132–10142.

Kingma, D. P.; Mohamed, S.; Rezende, D. J.; and Welling, M. 2014. Semi-supervised Learning with Deep Generative Models. In *NeurIPS*, 3581–3589.

Lake, B. M.; Salakhutdinov, R.; Gross, J.; and Tenenbaum, J. B. 2011. One shot learning of simple visual concepts. In *CogSci*.

Lake, B. M.; Salakhutdinov, R.; and Tenenbaum, J. B. 2015. Human-level concept learning through probabilistic program induction. *Science* 350(6266): 1332–1338.

Lee, K.; Maji, S.; Ravichandran, A.; and Soatto, S. 2019. Meta-Learning with Differentiable Convex Optimization. In *CVPR*, 10657–10665.

Li, F.-F.; Fergus, R.; and Perona, P. 2006. One-Shot Learning of Object Categories. *TPAMI* 28(4): 594–611.

Liu, Y.; Lee, J.; Park, M.; Kim, S.; Yang, E.; Hwang, S. J.; and Yang, Y. 2019. Learning to Propagate Labels: Transductive Propagation Network for Few-Shot Learning. In *ICLR*.

Maurer, A.; Pontil, M.; and Romera-Paredes, B. 2016. The Benefit of Multitask Representation Learning. *JMLR* 17: 81:1–81:32.

Mensink, T.; Verbeek, J. J.; Perronnin, F.; and Csurka, G. 2013. Distance-Based Image Classification: Generalizing to New Classes at Near-Zero Cost. *TPAMI* 35(11): 2624–2637.

Nagabandi, A.; Clavera, I.; Liu, S.; Fearing, R. S.; Abbeel, P.; Levine, S.; and Finn, C. 2019. Learning to adapt: Meta-learning for model-based control. *ICLR* .

Nichol, A.; Achiam, J.; and Schulman, J. 2018. On First-Order Meta-Learning Algorithms. *CoRR* abs/1803.02999.

Oliver, A.; Odena, A.; Raffel, C.; Cubuk, E. D.; and Goodfellow, I. J. 2018. Realistic Evaluation of Deep Semi-Supervised Learning Algorithms. In *NeurIPS*, 3239–3250.

Qiao, L.; Shi, Y.; Li, J.; Wang, Y.; Huang, T.; and Tian, Y. 2019. Transductive Episodic-Wise Adaptive Metric for Few-Shot Learning. In *ICCV*, 3603–3612.

Qiao, S.; Liu, C.; Shen, W.; and Yuille, A. L. 2018. Few-Shot Image Recognition by Predicting Parameters From Activations. In *CVPR*, 7229–7238.

Ravi, S.; and Larochelle, H. 2017. Optimization as a model for few-shot learning. In *ICLR*.

Ren, M.; Triantafillou, E.; Ravi, S.; Snell, J.; Swersky, K.; Tenenbaum, J. B.; Larochelle, H.; and Zemel, R. S. 2018. Meta-Learning for Semi-Supervised Few-Shot Classification. In *ICLR*.

Rusu, A. A.; Rao, D.; Sygnowski, J.; Vinyals, O.; Pascanu, R.; Osindero, S.; and Hadsell, R. 2018. Meta-Learning with Latent Embedding Optimization. In *ICLR*.

Saito, K.; Watanabe, K.; Ushiku, Y.; and Harada, T. 2018. Maximum Classifier Discrepancy for Unsupervised Domain Adaptation. In *CVPR*, 3723–3732.

Sajjadi, M.; Javanmardi, M.; and Tasdizen, T. 2016. Regularization With Stochastic Transformations and Perturbations for Deep Semi-Supervised Learning. In *NeurIPS*, 1163–1171.

Snell, J.; Swersky, K.; and Zemel, R. S. 2017. Prototypical Networks for Few-shot Learning. In *NeurIPS*, 4080–4090.

Sun, Q.; Liu, Y.; Chua, T.-S.; and Schiele, B. 2019. Meta-Transfer Learning for Few-Shot Learning. In *CVPR*, 403–412.

Triantafillou, E.; Zhu, T.; Dumoulin, V.; Lamblin, P.; Xu, K.; Goroshin, R.; Gelada, C.; Swersky, K.; Manzagol, P.-A.; and Larochelle, H. 2020. Meta-Dataset: A Dataset of Datasets for Learning to Learn from Few Examples. In *ICLR*.

Vilalta, R.; and Drissi, Y. 2002. A Perspective View and Survey of Meta-Learning. *Artificial Intelligence Review* 18(2): 77–95.

Vinyals, O.; Blundell, C.; Lillicrap, T.; Kavukcuoglu, K.; and Wierstra, D. 2016. Matching Networks for One Shot Learning. In *NeurIPS*, 3630–3638.

Xie, Q.; Dai, Z.; Hovy, E.; Luong, M.; and Le, Q. V. 2020. Unsupervised Data Augmentation for Consistency Training. In *NeurIPS*.

Ye, H.-J.; Hu, H.; Zhan, D.-C.; and Sha, F. 2020. Few-Shot Learning via Embedding Adaptation With Set-to-Set Functions. In *CVPR*, 8805–8814.

Ye, H.-J.; Lu, S.; and Zhan, D.-C. 2020. Distilling Cross-Task Knowledge via Relationship Matching. In *CVPR*, 12393–12402.

Yoon, S. W.; Seo, J.; and Moon, J. 2019. TapNet: Neural Network Augmented with Task-Adaptive Projection for Few-Shot Learning. In *ICML*, 7115–7123.

Yu, T.; Finn, C.; Dasari, S.; Xie, A.; Zhang, T.; Abbeel, P.; and Levine, S. 2018. One-Shot Imitation from Observing Humans via Domain-Adaptive Meta-Learning. In *Robotics: Science and Systems*.

Zhang, J.; Zhao, C.; Ni, B.; Xu, M.; and Yang, X. 2019. Variational Few-Shot Learning. In *ICCV*, 1685–1694.