Optimal Transport of Diverse Unsupervised Tasks for Robust Learning from Noisy Few-Shot Data Supplementary Material

Xiaofan Que and Qi Yu

Rochester Institute of Technology
{xq5054, qi.yu}@rit.edu

Organization of the appendix. In Appendix A, we summarize the major notations used in paper along with the pseudo code of the training process nof DUNT. Details of the datasets, and baselines are presented in Appendix B.1 and Appendix B.2, respectively. Additional ablation studies are presented in Appendix C. In Appendix D, we discuss potential societal impacts and limitations. The link to the source code is provided in Appendix E.

A Notation and Algorithm

Tab. 1 summarizes the major notations used in this work. The overall training process is depicted in Algorithm 1. The framework is generally applicable to any few-shot meta-learning methods.

B Additional Experimental Settings

B.1 Details of the Datasets

The details of the datasets are described as follows:

- *CUB* [51] contains 11,788 images of 200 bird species. Classes are split into 100, 50, and 50 for train, test, and validation, respectively. To mimic the real-world label noise, we further arrange different class-level noise ratios for different classes. Specifically, there are 10 classes with 50% noise, 10 classes with 33% noise, 10 classes with 16% noise, 40 classes with 8% noise, and the rest 30 classes are clean, resulting the final noise ratio to be around 12%.
- *miniImageNet (mini)* [48] contains 100 different classes with 600 images per class, which are split into 64, 20, and 16 for training, testing and validation, respectively. Similarly, we further arrange different class-level noise ratios for different classes. Specifically, there are 5 classes with 50% noise, 10 classes with 33% noise, 10 classes with 16% noise, 10 classes with 8% noise, 10 classes with 3% noise, 10 classes with 2% noise, and the rest 9 classes clean, resulting the final noise ratio to be around 14%.

Table 1: Notations and Descriptions

Notation	Description
$\mathbb{P}_X,\ \mathbb{P}_{X'}$	data distribution over random variable X, X'
$\prod(\mathbb{P}_X,\ \mathbb{P}_{X'})$	the set of all joint distributions whose marginals are $\mathbb{P}_X,\ \mathbb{P}_{X'}$
$\gamma(\mathbf{x}, \mathbf{x}')$	a coupling of random variables $\mathbf{x}, \mathbf{x}'$
$c(\mathbf{x}, \mathbf{x}')$	cost function for moving $\mathbf{x}$ to $\mathbf{x}'$
ξ	Wasserstein distance metric hyper-parameter
$\mathcal{D}^{\text{pri}}$	primary data distribution
$\tilde{\mathcal{D}}^{\text{pri}}$	noisy primary data distribution
$\mathcal{D}^{\text{aux}}$	auxiliary data distribution
$\mathcal{T}$	task
$\{\mathcal{T}_j\}^{\text{val}}$	valuable task group
$\{\mathcal{T}_j\}^{\text{conf}}$	confusing task group
θ	parameter of meta-model
θ_i'	parameter of task-specific model of task $\mathcal{T}_i$
θ^c	parameter of classification model
θ^d	parameter of critic model
w	episode cleansing weight parameter
v	task selection weight parameter
A_q	Augmentation q
η^d, η^c	learning rate for model updates
$\mathcal{S}_i^{\text{sup}}$	support set samples of $\mathcal{T}_i$
$\mathcal{S}_i^{\text{que}}$	query set samples of $\mathcal{T}_i$
$\hat{\mathcal{S}}_i^{\text{que}}$	clean query set samples of task $\mathcal{T}_i$
α_1, α_2	hyper-parameter of the total loss
γ_{th}	hyper-parameter of episode cleansing threshold
γ_{div}	hyper-parameter of task selection threshold

Algorithm 1 DUNT Training

INPUT initialized meta-model θ, θ^c, θ^d, α_1,α_2, task distributions $\tilde{\mathcal{D}}^{\text{pri}}$, $\mathcal{D}^{\text{aux}}$
while not done **do**
  Construct a batch of supervised tasks from the primary distribution $\{\mathcal{T}_i\} \sim \tilde{\mathcal{D}}^{\text{pri}}$
  for a mini-batch of tasks $\mathcal{T}_i$ **do**
    Update task-specific model θ_i' using data $\mathcal{S}_i^{\text{sup}}$
    Use θ_i' for episode cleansing query data from $\mathcal{S}_i^{\text{que}}$, and obtain the clean query set $\hat{\mathcal{S}}_i^{\text{que}}$
  end for
  Construct a batch of unsupervised tasks from the auxiliary distribution $\{\mathcal{T}_j\} \sim \mathcal{D}^{\text{aux}}$
  Separate valuable tasks $\{\mathcal{T}_j\}^{\text{val}}$ from confusing tasks $\{\mathcal{T}_j\}^{\text{conf}}$ using the diverseness metric
  Update meta-model parameter θ with $\{\mathcal{S}_i^{\hat{\text{que}}}\}$ and $\{\mathcal{T}_j\}^{\text{val}}$ using:

$$\theta \leftarrow \theta - \eta \left(\frac{\alpha_1}{|\mathcal{D}^{\text{pri}}|} \sum_{(\mathbf{x},y)\sim\mathcal{D}^{\text{pri}}} \frac{\partial \mathcal{L}^{\text{pri}}}{\partial \theta} + \frac{1-\alpha_1}{|\mathcal{D}^{\text{aux}}|} \sum_{(\mathbf{x},y)\sim\mathcal{D}^{\text{aux}}} \frac{\partial \mathcal{L}^{\text{aux}}}{\partial \theta} \right)$$
$$-\eta \left(\frac{\alpha_2}{|\mathcal{D}^{\text{pri}}|} \sum_{\mathbf{x}\sim\mathcal{D}^{\text{pri}}} \frac{\partial g(\mathbf{x})}{\partial \theta} - \frac{\alpha_2}{|\mathcal{D}^{\text{aux}}|} \sum_{\mathbf{x}\sim\mathcal{D}^{\text{aux}}} \frac{\partial g(\mathbf{x})}{\partial \theta} \right)$$

  Update classifier parameter θ^c using:

$$\theta^c \leftarrow \theta^c - \eta^c \left(\frac{\alpha_1}{|\mathcal{D}^{\text{pri}}|} \sum_{(\mathbf{x},y)\sim\mathcal{D}^{\text{pri}}} \frac{\partial \mathcal{L}^{\text{pri}}}{\partial \theta^c} + \frac{1-\alpha_1}{|\mathcal{D}^{\text{aux}}|} \sum_{(\mathbf{x},y)\sim\mathcal{D}^{\text{aux}}} \frac{\partial \mathcal{L}^{\text{aux}}}{\partial \theta^c} \right)$$

  Update critic parameter θ^d using:

$$\theta^d \leftarrow \theta^d + \eta^d \left(\frac{\alpha_2}{|\mathcal{D}^{\text{pri}}|} \sum_{\mathbf{x}\sim\mathcal{D}^{\text{pri}}} \frac{\partial g(\mathbf{x})}{\partial \theta^d} - \frac{\alpha_2}{|\mathcal{D}^{\text{pri}}|} \sum_{\mathbf{x}\sim\mathcal{D}^{\text{aux}}} \frac{\partial g(\mathbf{x})}{\partial \theta^d} \right)$$

end while

- *tieredImageNet (tiered)* [38] has 34 high-level categories and 608 sub-categories, each of which consists of around 1,300 images. These categories are split into 20 (351 sub-categories), 8 (160 sub-categories), and 6 (97 sub-categories) for training, testing, and validation, respectively. The noise ratio is 20%, and the noise is distributed equally in each training classes.
- *CIFAR-FS (CF)* [3] includes 100 different classes with 600 images/class. These classes are further grouped into 20 super-classes. The 100 classes are split into 64, 20, 16 for train, test, and validation, respectively. The noise ratios is 10%, and the noise is distributed equally in each training classes.
- *Cross-Domain* where the training classes are from miniImageNet dataset and the validation and testing classes are from the CUB, EuroSAT, CropDisease, ISIC, and ChestX [16] datasets, respectively.

B.2 Detailed Settings and Descriptions of the Baselines

For a fair comparison, all methods in our implementation use the TorchMeta package [9] with default parameters. All the experiments are conducted on a NVIDIA A100 GPU. The details of the baselines are described as follows:

- PN+SPL: we implement ProtoNet with a data cleansing method SPL [26] that uses the small-loss trick to remove noisy data.
- PN+FSR: we implement ProtoNet with a data cleansing method [61] that removes noisy samples by large margin strategy.
- PN+Co/PN+Co+: we implement CoTeaching [17] and CoTeaching+ [59] by training both networks as ProtoNet and we also use small-loss trick to remove noisy data. For a fair comparison, we utilize the transductive setting where the peer networks are trained on the primary and auxiliary data, respectively.
- STARTUP [35] is a transductive FSL method with unlabeled data for relaxing the extreme domain gap between train and test. Since it's not explicitly designed for the noisy labels, for a fair comparison, we use pre-trained TraNFS as the teacher model to relabel the unlabeled test data and then train the student model with meta-train data and pseudo-label meta-test data.
- DDNet [21] solves cross-domain few-shot learning with the proposed distillation loss. Similar to STARTUP, we also use pre-trained TraNFS as the teacher model to relabel the unlabeled images.
- RNNP [31] is a noisy few-shot learning method that uses hybrid features in ProtoNets to learn a model that is robust to noisy labels.
- TraNFS [28] uses weighted prototypes for learning a model robust to label noise. In our implementation, the spatial median prototypes are utilized based on ProtoNets.
- DCML [37] uses dual-level curriculum learning and episode cleansing to remove noisy data in few-shot learning tasks.

C Additional Ablation Study

In this section, we present the study of the impact of different auxiliary datasets, and additional ablation studies about (1) the impact of the noise ratio; (2) the impact of the noise types; (3) the impact of threshold γ_{th}; (4) the impact of hyperparameter α_1 and α_2; (5) the impact of different regularization terms; (6) the impact of different auxiliary tasks; and finally, we provide DUNT framework based on (7) other meta-learning model: FOMAML.

MiniImageNet with different auxiliary datasets. In Tab. 2, we have tested the impact of auxiliary datasets on MiniImagenet by training the models on the training set of MiniImagenet with different auxiliary datasets. We have tested with all the available datasets in this work along with the unlabeled MiniImagenet test data. The results shows the more similar the auxiliary dataset to the MiniImagenet data, the better the performance is. mini-test ranks the highest over all since they are sampled from the same domain, which shows the superiority of transductive setting. Other factors such as auxiliary dataset size, dataset closeness also have some impact on the performance of the primary task. All in all, the performance of `DUNT` oscillate less as compared to other methods, showing that it is capable of balancing the contribution of the primary and the auxiliary flexibly, reducing the negative impact of various factors.

Table 2: MiniImageNet with different auxiliary datasets

Methods	CUB	tiered	CIFAR-FS	EuroSAT	CropDisease	ChestX	ISIC	mini-test
Task	5-way 1-shot							
PN+Co	47.79 ± 1.33	48.07 ± 1.41	46.20 ± 1.24	50.70± 1.23	47.60 ± 2.11	50.17 ± 1.19	49.24 ± 1.27	44.44 ± 1.25
PN+Co+	47.16 ± 1.21	45.44 ± 1.32	45.27 ± 0.91	47.25 ± 0.83	49.48 ± 1.11	49.64 ± 1.11	50.01 ± 0.81	45.08 ± 0.83
STARTUP	48.18 ± 1.31	48.97 ± 1.44	49.24 ± 0.92	46.82 ± 1.11	48.66 ± 1.13	46.02 ± 1.20	39.99 ± 1.26	48.79 ± 1.19
DDNet	48.94 ± 1.09	50.01 ± 1.09	49.89 ± 0.81	49.29 ± 0.99	49.79 ± 1.09	50.11 ± 1.08	44.32 ± 1.61	50.03 ± 1.23
DUNT	**49.96** ± 1.08	**50.27** ± 1.02	**51.31** ± 1.02	**51.09** ± 1.20	**50.89** ± 1.04	**51.66** ± 0.94	**50.44** ± 1.01	**51.64** ± 1.01
Task	5-way 5-shot							
PN+Co	61.30 ± 1.13	61.29 ± 2.14	58.49 ± 1.22	**65.72** ± 1.14	60.59 ± 1.30	64.42 ± 1.40	61.22 ± 1.11	59.57 ± 1.19
PN+Co+	59.47 ± 1.11	57.95 ± 1.09	58.04 ± 0.98	60.55 ± 1.20	63.92 ± 1.10	62.44 ± 1.08	62.12 ± 1.20	60.43 ± 1.15
STARTUP	63.22 ± 1.09	64.53 ± 1.13	66.49 ± 1.10	62.27 ± 1.20	61.32 ± 1.27	60.69 ± 1.30	54.40 ± 1.21	62.00 ± 1.17
DDNet	62.89 ± 1.00	65.45 ± 1.09	**67.66** ± 0.90	63.37 ± 1.32	64.01 ± 1.09	64.00 ± 1.01	63.39 ± 1.01	64.70 ± 1.10
DUNT	**64.86** ± 0.98	**66.14** ± 1.04	66.34 ± 1.23	65.10 ± 1.30	**64.76** ± 1.21	**64.73** ± 1.32	**65.01** ± 1.21	**66.28** ± 1.20

Impact of noise ratio. In this ablation, we study the impact of noisy level in CIFAR-FS. We've tested with 10% to 50% of label noise and the experimental results are shown in Fig. 1. From the results, we can observe: i) `DUNT` ranks the top in all different noise ratios; and ii) as the noise level increases, the performance of all methods decrease drastically while our method has the slowest decreasing rate. The observations conclude that `DUNT` has a stable performance despite the changing noise level and is robust to the attack.

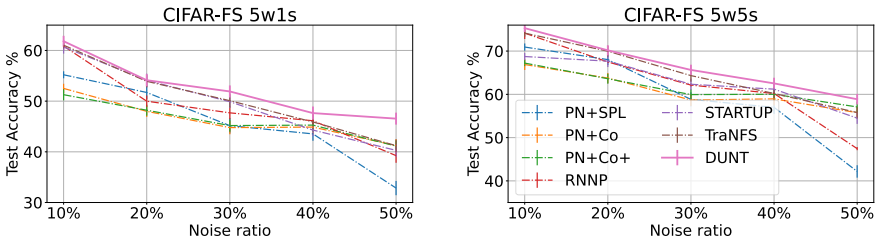


Fig. 1: Test accuracy of DUNT with different noise ratios: **Left:** 5-way 1-shot; **Right:** 5-way 5-shot.

Impact of label noise type. In Tab. 1 to Tab. 2, the class-level symmetric label noise is synthesized. In this ablation, we consider asymmetric label noise and real-world label noise. As shown in the left figure of Fig. 2, we conduct 5-way 1-shot experiment on CIFAR-FS dataset with 10% to 50% of asymmetric label noise, which follows the same synthetic asymmetric label corruption as in [17]. It is noted that asymmetric label noise has a more pronounced detrimental effect on model performance. As the noise ratio increases, there is a significant decline in the model's effectiveness. Remarkably, our framework surpasses all baseline models across various noise ratios, demonstrating superior performance. Notably, our model maintains stability even when confronted with an exceptionally high 50% label noise. In the right figure of Fig. 2, we show the 5-way 1-shot results of the real-world noisy dataset CIFAR-100N[1]. In the evaluation phase, synthetic symmetric label noise is introduced into the support set of each task, with variations of 20% and 40%. Additionally, we include a scenario with 0% synthetic noise in the support set during evaluation for comparative analysis. It is observed that, our method demonstrates minimal degradation in model performance even when synthetic noise is introduced to the support set, unlike other approaches where the addition of synthetic noise significantly impairs performance.

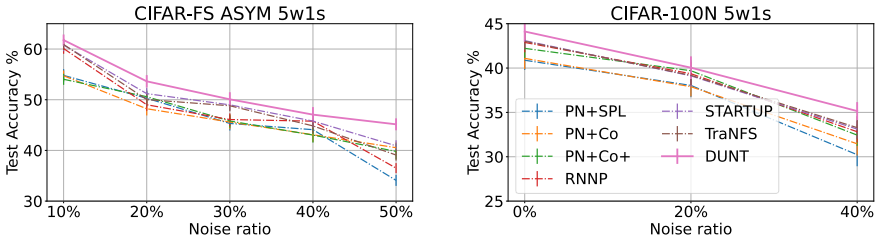


Fig. 2: Test accuracy of DUNT with different noise types : **Left:** Asymmetric label noise; **Right:** Real-world label noise.

[1] https://github.com/UCSC-REAL/cifar-10-100n/tree/main

Additionally, we've tested the episode cleansing for the support set label noise on 5-way 5-shot miniImageNet dataset, where we add 20% and 40% synthetic label noise to the support set during evaluation, respectively. From the result shown in Tab. 3, we can conclude that episode cleansing strategy is very effective to support set label noise.

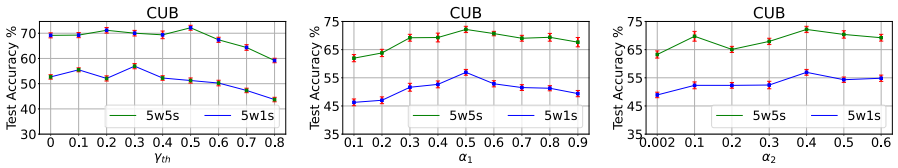
Table 3: Result on MiniImageNet with support set label noise during evaluation

DUNT	20%	40%
w/ EC	0.3544	0.3300
w/o EC	0.5751	0.4927

Impact of threshold γ_{th}. In this experiment, we test the effect of the hyperparameter threshold γ_{th} in the weight assignment rule on CUB dataset with synthetic noise. According to [2, 26], we set the value of γ_{th} by the percentage of noisy examples removed and its value range from 0% to 80%, which is a common practice of self-paced learning approaches. When $\gamma_{th} = 0$, no noisy data is removed and we obtain the vanilla DUNT framework. As shown in Fig. 3a, $\gamma_{th} = 30\%$ achieves the highest performance for 5-way 1-shot experiment (please refer to the Appendix), and $\gamma_{th} = 60\%$ achieves the highest performance for the 5-shot one. The performance of vanilla `DUNT` is close to the best performance, showing the effectiveness of auxiliary data and the domain-invariant representation learning. On the other hand, the general trend shows that removing too many examples harms the model's performance, which confirms the hypothesis of information loss over data cleansing.

Impact of hyperparameters α_1 and α_2. In this experiment, we test the effect of α_1 and α_2 used in the objective Eq. (7) on CUB dataset. In Fig. 3b the value of α_1 is set among 0.1 to 0.9, with α_2 fixed as 0.2. The results show that using relatively balanced hyperparameters from 0.4 to 0.6, achieves higher performance, which means extra knowledge from the auxiliary task helps the generalization of the model with proper incorporation with the primary data. On the contrary, setting α_1 either to a very small value or a very high value can not achieve a desirable result since the balanced contribution matters in `DUNT`. In Fig. 3c, when we test the impact of α_2, we keep α_1 fixed as 0.5 while setting the values of α_2 are set as 0.002, and from 0.1 to 0.6. The results show that a very small α_2 value doesn't work as well as the relatively larger ones considering the divergence between the primary and auxiliary distribution should be constrained. As for other values, the performance of 5-way 1-shot task are all over 50%, showing a consistently high performance.

Impact of regularization terms. OT regularization aims to minimize the discrepancy between the primary and the auxiliary tasks distributions. In this ablation study, we verify the effectiveness of OT regularization by removing this component, and test other two types of regularization: KL-divergence (*i.e.* KLD) and Maximum Mean Discrepancy (*i.e.* MMD). Both are used to measure the discrepancy between two domains. We conduct 5w1s and 5w5s experiments on five different datasets under transductive setting, and the results are shown in Tab. 4. The results on cross-domain dataset are shown in Tab. 5. First of all, with the regularization term, `DUNT` clearly performs better, which verifies the



(a) **(b)** **(c)**

Fig. 3: Test accuracy of DUNT on CUB with different γ_{th} (a), α_1 (b), α_2 (c).

proposition that by minimizing the domain gap, the model learns a better domain invariant representation. Besides, Wassertein distance outperforms MMD and KLD in most cases and shows a consistently stable and high performance, which shows the superiority of Wassertein distance based optimal transport.

Table 4: Test accuracy of different regularization terms on transdutive datasets

Datasets	CUB	mini	mini-CUB	tiered	CIFAR-FS
Methods	5-way 1-shot				
Without Reg	53.55 ± 1.15	50.02 ± 1.09	40.04 ± 1.15	45.30 ± 1.10	56.25 ± 1.11
MMD	54.88 ± 1.23	49.60 ± 0.88	40.56 ± 0.96	45.91 ± 1.17	38.57 ± 1.05
KLD	**55.86** ± 1.09	50.85 ± 1.02	39.91 ± 0.88	46.29 ± 1.03	40.39 ± 1.04
DUNT	55.01 ± 1.12	**52.38** ± 0.85	**42.07** ± 1.10	**49.29** ± 1.26	**61.90** ± 1.14
Methods	5-way 5-shot				
Without Reg	67.33 ± 0.78	62.55 ± 0.91	53.28 ± 0.90	59.03 ± 1.10	70.22 ± 1.09
MMD	68.67 ± 0.91	62.52 ± 0.83	53.78 ± 1.02	59.00 ± 1.00	47.17 ± 0.93
KLD	69.39 ± 0.89	64.93 ± 0.88	52.95 ± 1.03	58.93 ± 1.04	51.39 ± 0.92
DUNT	**71.20** ± 1.12	**66.46** ± 0.98	**59.05** ± 1.01	**61.09** ± 0.90	**76.11** ± 0.98

Impact of auxiliary tasks. Besides the unsupervised auxiliary task, we conduct two more self-supervised learning (SSL) tasks: image rotations (Rotation) and relative patch location (PatchLocation) [13] as auxiliary tasks. Tab. 6 and Tab. 7 shows a performance drop as compared to unsupervised classification task. The main reason is that by utilizing extra unlabeled data as unsupervised classification task, we manage to compensate the information loss caused by data cleansing in noisy few-shot learning. While the SSL tasks can help the model learn a better feature representation, it fails to keep up with the advantage of using the extra information of the unlabeled data.

Generalization to other meta-learning models. The above experiments are conducted based on one of the most classic meta-learning method ProtoNet [43], in which our model has shown superior performance. Moreover, our framework is model agnostic and it can be easily generalized to other types of meta-learning methods. We provide a DUNT implementation based on another classic few-shot

Table 5: Test accuracy of different regularization terms on cross-domain datasets

Datasets	EuroSAT	CropDisease	ChestX	ISIC
Tasks	5-way 5-shot			
Without Reg	53.93 ± 1.19	56.77 ± 1.09	27.13 ± 0.63	33.23 ± 0.70
MMD	57.85 ± 1.89	57.10 ± 1.22	28.06 ± 0.73	34.63 ± 0.85
KLD	56.08 ± 2.37	59.13 ± 1.23	27.69 ± 0.68	30.21 ± 0.78
DUNT	**57.10** ± 1.29	**61.49** ± 1.01	**29.78** ± 0.90	**35.80** ± 0.77
Tasks	5-way 5-shot			
Without Reg	70.10 ± 1.41	77.67 ± 0.74	30.11 ± 0.68	37.10 ± 1.65
MMD	**73.09** ± 1.33	76.86 ± 0.80	30.86 ± 0.79	38.11 ± 1.60
KLD	71.39 ± 1.52	78.66 ± 0.91	30.21 ± 0.73	37.68 ± 1.99
DUNT	72.81 ± 1.10	**83.20** ± 1.08	**33.94** ± 1.20	**40.31** ± 0.90

Table 6: Test accuracy of different auxiliary tasks on different datasets

SSL tasks		CUB	mini	mini-CUB	tiered	CIFAR-FS
PatchLocation	5w1s	54.77 ± 1.23	48.16 ± 0.49	40.25 ± 0.80	48.10 ± 1.00	59.55 ± 0.72
Rotation		54.86 ± 1.19	49.56 ± 0.43	41.53 ± 0.60	47.78 ± 1.30	60.45 ± 0.31
DUNT		**56.94** ± 1.03	**52.22** ± 1.10	**43.01** ± 0.51	**49.12** ± 0.68	**62.80** ± 0.49
PatchLocation	5w5s	68.44 ± 0.90	63.89 ± 0.39	58.12 ± 0.62	60.00 ± 1.09	73.33 ± 0.89
Rotation		69.00 ± 0.89	63.70 ± 0.36	57.29 ± 0.37	59.39 ± 1.11	71.79 ± 0.24
DUNT		**72.17** ± 1.11	**65.10** ± 1.23	**58.50** ± 1.11	**62.01** ± 1.09	**74.44** ± 1.42

Table 7: Test accuracy of different auxiliary tasks on cross-domain datasets

Datasets		EuroSAT	CropDisease	ChestX	ISIC
PatchLocation	5w1s	55.80 ± 1.08	59.23 ± 1.01	26.77 ± 0.80	33.03 ± 1.08
Rotation		55.08 ± 1.07	59.88 ± 1.33	27.80 ± 1.09	30.19 ± 1.78
DUNT		**57.10** ± 1.29	**61.49** ± 1.01	**29.78** ± 0.90	**35.80** ± 0.77
PatchLocation	5w5s	71.09 ± 1.09	81.80 ± 1.22	31.01 ± 0.88	39.12 ± 1.18
Rotation		70.90 ± 1.00	80.60 ± 1.09	32.02 ± 1.20	39.11 ± 1.78
DUNT		**72.81** ± 1.10	**83.20** ± 1.08	**33.94** ± 1.20	**40.31** ± 0.90

Table 8: Non-transductive auxiliary datasets

Datasets	CUB	mini	mini-CUB	tiered	CIFAR-FS
Tasks	5-way 1-shot				
Without Reg	54.61 ± 1.08	49.09 ± 1.01	38.92 ± 0.87	43.44 ± 1.09	55.89 ± 0.98
MMD	55.76 ± 1.27	48.75 ± 0.90	39.52 ± 0.92	43.61 ± 1.10	40.96 ± 1.05
KLD	56.37 ± 1.07	49.04 ± 0.99	41.48 ± 0.78	42.07 ± 1.23	41.66 ± 1.19
DUNT	**56.94** ± 1.03	**52.22** ± 1.10	**43.01** ± 0.51	**49.12** ± 0.68	**62.80** ± 0.49
Tasks	5-way 5-shot				
Without Reg	70.01 ± 1.01	62.43 ± 0.90	51.09 ± 0.88	52.66 ± 1.07	70.09 ± 1.11
MMD	69.13 ± 0.95	63.33 ± 1.06	51.61 ± 0.92	54.94 ± 1.06	52.62 ± 1.01
KLD	70.91 ± 1.06	62.48 ± 0.84	57.20 ± 0.96	53.41 ± 1.05	52.85 ± 1.09
DUNT	**72.17** ± 1.11	**65.10** ± 1.23	**58.50** ± 1.11	**62.01** ± 1.09	**74.44** ± 1.42

learning method, FOMAML [10], and test on different datasets with the same settings as mentioned before. The parameters of the base model and our method are set the same as [10]. From Tab. 9's result, solid performance and advantage over the baseline are observed, which confirms the generalization of `DUNT` to other types of meta-learning models.

Table 9: Few-shot performance based on FOMAML

Datasets	CUB	mini	mini-CUB	tiered	CF
Task	5-way 1-shot				
FOMAML	45.11 ± 1.05	33.94 ± 1.12	31.93 ± 1.11	25.95 ± 1.15	48.49 ± 1.03
DUNT	**45.41** ± 1.15	**39.61** ± 0.95	**35.09** ± 1.01	**34.89** ± 1.22	**50.95** ± 1.20
Task	5-way 5-shot				
FOMAML	51.06 ± 1.15	37.52 ± 1.15	37.15 ± 1.15	26.11 ± 1.15	58.53 ± 1.15
DUNT	**52.93** ± 1.33	**47.01** ± 1.21	**43.31** ± 1.30	**44.51** ± 0.91	**62.52** ± 1.05

D Potential Societal Impacts and Limitations

Potential societal impacts. Few-shot learning can make it easier to develop machine learning models with limited labeled data. This could potentially democratize access to technology, allowing a broader range of individuals and communities to benefit from machine learning applications. If the few-shot learning process is trained on biased or noisy data, it can perpetuate and even exacerbate existing biases. This may result in unfair or discriminatory outcomes, impacting marginalized groups and reinforcing societal inequalities. This work aims at

solving the noisy few-shot learning problem and making it possible for using machine learning and deep learning techniques on data scarce research, such as the medical images/data analysis of some rare diseases.

Limitations and potential extensions. In our current theoretical analysis, we only consider the relationship between the primary and auxiliary distributions, and the generalization bound. For potential extension, this work opens interesting directions for future research: (i) When training deep neural networks with the auxiliaries, we observe better performance in terms of generalization bound, which can be further measured and quantified. (ii) When the noise is considered in the theoretical analysis, a natural question is that what can be done to reduce the negative impact according to the theoretical insights.

E Source Code

Source code is available at `https://github.com/ritmininglab/DUNT`.