Learning to Learn Semantic Factors in Heterogeneous Image Classification

Boyue Fan
University of Sheffield
United Kingdom

Zhenting Liu
Pasadena City college
United States

Abstract

Few-shot learning is to recognize novel classes with a few labeled samples per class. Although numerous meta-learning methods have made significant progress, they struggle to directly address the heterogeneity of training and evaluating task distributions, resulting in the domain shift problem when transitioning to new tasks with disjoint spaces. In this paper, we propose a novel method to deal with the heterogeneity. Specifically, by simulating class-difference domain shift during the meta-train phase, a bilevel optimization procedure is applied to learn a transferable representation space that can rapidly adapt to heterogeneous tasks. Experiments demonstrate the effectiveness of our proposed method.

1 Introduction

Deep learning methods are now widely used in diverse applications. However, their efficacy is largely contingent on a large amount of labelled data in the target task and domain of interest (Vaswani et al., 2017). Different from humans that can easily learn to accomplish new tasks with a few examples, it is difficult for machines to rapidly generalize to new concepts with very little supervision, which calls considerable attention to the challenging few-shot learning (FSL) setting. For example, few-shot classification problem requires models to classify unlabeled samples into novel classes with only a few labeled samples available for training (Finn et al., 2017). Commonly understood as learning to learn, meta-learning paradigm has made significant progress in FSL by transferring knowledge extracted from a collection of previous tasks (Vinyals et al., 2016; Snell et al., 2017). Such task-agnostic knowledge can contribute to the current testing task with optimizing learning algorithms. However, beyond its recent achievements, meta-learning still faces the problem of generalization.

In contrast to supervised machine learning methods which assume that training and testing data are sampled i.i.d. from the same distribution, FSL aims to learn to address tasks from different distributions with limited data. This refers to the realistic scenario that the label spaces of future testing tasks can not be obtained in advance and are often disjoint with the label spaces of training tasks. In experiments, this is actualized by splitting all categories in the dataset into non-overlapping base classes and novel classes, while training tasks are sampled from base classes and testing tasks are samples from novel classes. Therefore, due to the class label difference, meta-learning approaches suffer from natural heterogeneous distributions of tasks. As each task can be regarded as having a separate domain, it can be considered as a special case of domain shift that is extremely serious when a large gap of semantic relationship exists between base classes and novel classes.

As most of the current meta-learning approaches make a strong assumption that training tasks and testing tasks are drawn from the similar distributions and share the same characteristics, (Chen et al., 2019) has shown the limitations of existing approaches in cross-domain FSL scenarios where base classes and novel classes are from different datasets. However, few works have focused on this issue to improve existing approaches. For example, as a representative work of metric-based meta-learning, Prototypical Network (Snell et al., 2017) learns a metric space where embeddings of query samples in one class are close to the centroid of support samples in the same class, and far from centroids of other classes in the task. While Prototypical Network benefits from a simple but effective inductive bias, it lacks adaptation to new tasks or domains.

In this paper, we propose to improve such metric-based approaches with a bilevel optimization procedure. Specifically, we simulate class-difference-caused domain shift during meta-training by simultaneously sampling multiple tasks with non-

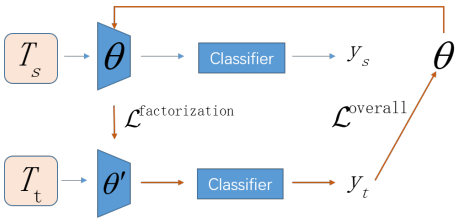

Figure 1: Overview of our proposed Meta-ProtoNet.

overlapping class sets. Each time one of the tasks is prepared as the target task for outer level optimization and the others are first used as the source tasks for inner level optimization of the network. Following this training strategy during the meta-train phase, the model can better adapt to the testing tasks from heterogeneous distributions with an adaptation step.

Moreover, different from some usual options of inner objective, we use Shannon entropy as an unsupervised factorization loss to constrain the learned representations as near-binary codes (Chang et al., 2019). This can be viewed as learning a discriminative latent factor space for each task where each factor can be interpreted as a latent attribute that is corresponding to abstract visual concepts.

To summarize, our main contributions are :1) considering the challenge of heterogeneous task distributions faced by few-shot learning, we simulate the class-difference-caused domain shift in the meta-train phase, and devise a metric-based meta-learning approach integrated with a bilevel optimization for better generalization; 2) we propose to utilize an unsupervised factorization loss as the inner objective, making representations to be near-binary codes that reduce the difficulty of classifier learning. Meanwhile, due to the bilevel optimization between heterogeneous few-shot tasks during meta-training, the model can rapidly learn the representation space for testing tasks; 3) We conduct extensive experiments and analysis to demonstrate that our approach effectively improves the performance and interpretability under both conventional and cross-domain few-shot settings without introducing additional architectures, and thus it can be regarded as a better baseline.

2 Methodology

2.1 Prototypical Network.

As a simple but effective model for FSL learning, Prototypical Network (ProtoNet) (Snell et al., 2017) use an embedding function f_θ with parameters θ to encode each sample into a representation vector. For each class c in the class set C of the task T , a prototype vector p_c is defined as the mean vector of the embedded support samples in the class, which can be expressed as $p_c = \frac{1}{|\mathcal{S}_c|} \sum_{(x_i, y_i) \in \mathcal{S}_c} f_\theta(x_i)$. When inferring, the probability over classes for a query sample x_i is a softmax over the inverse of squared Euclidean distances between the query representation and prototype vectors, expressed as $P_\theta(y_i = c \mid x_i) = \frac{\exp\left(-\|f_\theta(x_i) - p_c\|^2\right)}{\sum_{c' \in C} \exp\left(-\|f_\theta(x_i) - p_{c'}\|^2\right)}$. The classification loss is the sum of negative log-probability of each query sample in task T with its ground-truth class label: $\mathcal{L}^{\text{classification}}(\theta) = -\sum_{c \in C} \sum_{x_i \in \mathcal{Q}_c} \log P_\theta(y_i = c \mid x_i)$.

2.2 Learning Latent Factors

As the embedding function f_θ of Prototypical Network can be any deep neural network, it is often organized as a convolutional neural network (CNN) for image classification tasks. In our MetaProtoNet, we set the activation function of the last layer to Sigmoid function $\sigma(x) = \frac{1}{1+\exp(-x)}$ instead of the most commonly used ReLU function. This limits the scale of the learned representations $f_\theta(x_i) \in (0, 1)^d$, where d denotes the dimension number of the representations. Deep architectures are capable of learning to extract useful infor mation from the samples, and potentially construct representations as the composition of the local abstract concepts that are useful for downstream tasks. Therefore, Sigmoid activated outputs of f_θ can be viewed as multi-label predictions on latent factors, as the activation of each dimension closer to 0 or 1 can be interpreted as the corresponding visual attributes being present and absent. Moreover, Meta-ProtoNet constrains the learned representations to become near-binary codes by applying Shannon entropy as an unsupervised factorization loss, expressed as

$$\mathcal{L}^{\text{factorization}}(\theta) = -\sum_{x_i \in \{\mathcal{S}, \mathcal{Q}\}} \langle f_\theta(x_i), \log(f_\theta(x_i)) \rangle \tag{1}$$

where $\log(\cdot)$ is applied element-wise, and $\langle \cdot, \cdot \rangle$ denotes the vector inner product operation. This not only encourages the representations to become more interpretable but also decreases the uncertainty of latent factors discovery.

2.3 Training Meta-ProtoNet

According to (Snell et al., 2017), Prototypical Network can be re-interpreted as a linear classifier that is applied to the representations learned by the nonlinear embedding function. With the improvement above, near-binary representations generated by the embedding function are expected to be preferable for the jointly learned linear classifier without sacrificing representation power and differentiable optimization for exactly binary codes (Li et al., 2017). However, it would result in a suboptimal representation space for heterogeneous testing tasks since the metric-based approach is no longer updated to adapt to new domains in the meta-test phase. To overcome the approaching domain shift problem, we devise a bilevel optimization procedure for a fast adaptation to the feature distribution in the new task.

Specifically, instead of randomly sampling a single task, we simultaneously sample m tasks $\mathcal{T}_{\text{set}} = \{\mathcal{T}_1, \cdots, \mathcal{T}_m\}$ without class overlap from the distribution over training tasks $p\left(\mathcal{T}^{tr}\right)$ in the metatrain stage. For each task in $\mathcal{T}_{\text{set}}$, we first denote it as the target task $\mathcal{T}_t$ and obtain a copy of the model parameters θ as θ', then θ' is updated by minimizing the factorization loss over each task $\mathcal{T}_s$ in the source tasks $\mathcal{T}_{\text{set}} - \mathcal{T}_t$. Each update of θ' can be expressed as

$$\theta' = \theta' - \alpha \nabla_{\theta'} \mathcal{L}^{factorization}\left(\theta'\right) \tag{2}$$

where α is the inner learning rate. This is viewed as the inner level of the bilevel optimization procedure, and after all of $\mathcal{T}_s$ are used for the update of θ', we utilize $\mathcal{T}_t$ to optimize the model. Specifically, the model parameters θ are updated as follows:

$$\theta = \theta - \beta \nabla_{\theta} \mathcal{L}^{\text{overall}}\left(\theta'\right) \tag{3}$$

where β is the outer learning rate. The meta-optimization is performed over the model parameters θ, whereas the objective $\mathcal{L}^{\text{overall}}\left(\theta'\right)$ is computed using the updated model parameters θ' and can be expressed as

$$\mathcal{L}^{\text{overall}}\left(\theta'\right) = \mathcal{L}^{\text{classification}}\left(\theta'\right) + \gamma \mathcal{L}^{\text{factorization}}\left(\theta'\right) \tag{4}$$

where γ is the trade-off hyperparameter. The key idea underlying the algorithm is that to alleviate the class-difference-caused domain shift, the task-specific knowledge including semantic information of categories is decomposed into reusable low-level task-agnostic knowledge by transferring latent factors across heterogeneous tasks. Each round of bilevel optimization can be viewed as a simulation of the whole process including meta-train and meta-test: In the inner level (corresponding to the meta-train phase), we encourage the model to learn to generate latent factors for tasks drawn from the source distribution. As high performance of classification on these tasks is not necessary and may be detrimental to the classification of heterogeneous target tasks, the inner objective only aims to discover latent factors and does not include classification loss. Moreover, we expect the learned latent factor space to be transferable, and thus the learning process of the source tasks can promote the learning of heterogeneous tasks. Therefore, in the outer level (corresponding to the meta-test phase), the model is optimized with the overall loss including classification loss and factorization loss.

2.4 Testing Meta-ProtoNet

In the meta-test phase, when adapting to each new testing task $\mathcal{T}_j$, the trained parameters θ are updated to θ' using only one gradient descent step with the factorization loss over $\mathcal{T}_j$. Therefore, a task-specific latent factor space of $\mathcal{T}_j$ is learned. The evaluation metric (i.e., the classification accuracy) is calculated with the updated parameters θ'.

3 Experiments

Datasets. In this paper, we address the few-shot classification problem under both conventional and cross-domain FSL settings. These settings are conducted on three benchmark datasets: miniImageNet (Vinyals et al., 2016), Caltech-UCSD-Birds 200-2011 (CUB) (Wah et al., 2011), and SUN Attribute Database (SUN) (Patterson et al., 2014).

Experimental Settings. We conduct experiments on 5-way 1-shot and 5-way 5-shot settings, there are 15 query samples per class in each task. We report the average accuracy (%) and the corresponding 95% confidence interval over the 2000 tasks randomly sampled from novel classes. To fairly evaluate the original performance of each method, we use the same 4-layer ConvNet (Vinyals et al., 2016) as the backbone for all methods and do not adopt any data augmentation during training. All methods are trained via SGD with Adam (Kingma and Ba, 2014), and the initial learning rate is set to e^{-3}. For each method, models are trained for 40,000 tasks at most, and the best model on the vali-

Method	miniImageNet → CUB		miniImageNet → SUN		CUB → miniImageNet	
	5-way 1-shot	5-way 5-shot	5-way 1-shot	5-way 5-shot	5-way 1-shot	5-way 5-shot
Meta-Learner LSTM	23.77	30.58	25.52	32.14	22.58	28.18
MAML	40.29	53.01	46.07	59.08	33.36	41.58
Reptile	24.66	40.86	32.15	50.38	24.56	40.60
Matching Network	38.34	47.64	39.58	53.20	26.23	32.90
Prototypical Network	36.60	54.36	46.31	66.21	29.22	38.73
Relation Network	39.33	50.64	44.55	61.45	28.64	38.01
Baseline	24.16	32.73	25.49	37.15	22.98	28.41
Baseline++	29.40	40.48	30.44	41.71	23.41	25.82
Meta-ProtoNet	**40.61**	**56.12**	**49.38**	**68.80**	**33.58**	**43.83**

Table 1: Average accuracy (%) comparison to state-of-the-arts with 95% confidence intervals on 5-way classification tasks under the cross-domain FSL setting. Best results are displayed in boldface.

Method	miniImageNet		CUB		SUN	
	5-way 1-shot	5-way 5-shot	5-way 1-shot	5-way 5-shot	5-way 1-shot	5-way 5-shot
Meta-Learner LSTM	24.99	29.79	36.23	44.39	30.99	44.86
MAML	45.69	60.90	48.87	63.99	57.75	71.45
Reptile	26.59	39.87	27.21	42.35	28.30	51.62
Matching Network	47.63	56.28	53.06	62.19	55.02	62.57
Prototypical Network	46.15	65.56	48.21	57.80	55.70	67.32
Relation Network	47.64	63.65	52.76	64.71	58.29	72.15
Baseline	23.84	32.09	25.14	35.35	27.44	34.54
Baseline++	30.15	41.19	32.48	42.43	35.56	44.42
Meta-ProtoNet	**47.87**	**66.05**	**53.30**	**65.37**	**58.79**	**73.90**

Table 2: Average accuracy (%) comparison to state-of-the-arts with 95% confidence intervals on 5-way classification tasks under the conventional FSL setting. Best results are displayed in boldface.

dation classes is used to evaluate the final reporting performance in the meta-test phase.

Evaluation Using the Conventional Setting. Table 1 shows the comparative results under the conventional FSL setting on three benchmark datasets. It is observed that Meta-ProtoNet outperforms the original Prototypical Network in all conventional FSL scenarios. For 1-shot and 5-shot on miniImageNet → miniImageNet, Meta-ProtoNet achieves about 1% higher performance than Prototypical Network. However, Meta-ProtoNet achieves 5% and 10% higher performance for 1-shot and 5-shot on CUB → CUB, and 3% and 6% higher performance on SUN → SUN. As the latter two scenarios are conducted on fine-grained classification datasets, we attribute the promising improvement to that the categories in these fine-grained datasets share more local concepts than those in coarse-grained datasets, and thus a more discriminative space can be rapidly learned with a few steps of adaptation. Moreover, Meta-ProtoNet achieves the best performance among all baselines in all conventional FSL scenarios, which shows that our approach can be considered as a better baseline option under the conventional FSL setting.

Evaluation Using the Cross-Domain Setting. We also conduct cross-domain FSL experiments and report the comparative results in Table 2. Compared to the results under the conventional setting, it can be observed that all approaches suffer from a larger discrepancy between the distributions of training and testing tasks, which results in a performance decline in all scenarios. However, Meta-ProtoNet still outperforms the original Prototypical Network in all cross-domain FSL scenarios, demonstrating that the bilevel optimization strategy for adaptation and the learning of transferable latent factors can be utilized to improve simple metric-based approaches. Also, Meta-ProtoNet achieves all the best results, indicating that our approach can be regarded as a promising baseline under the cross-domain setting.

4 Conclusion

In this paper, we propose Meta-ProtoNet to handle the challenge of heterogeneous task distributions in few-shot scenarios, aiming to learn a latent factor space in which metric-based classification of heterogeneous tasks can be better performed. Extensive experiments show that our proposed approach can be considered as a stronger baseline in both conventional and cross-domain few-shot settings.

References

Xiaobin Chang, Yongxin Yang, Tao Xiang, and Timothy M Hospedales. 2019. Disjoint label space transfer learning with common factorised space. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3288–3295.

Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. 2019. A closer look at few-shot classification. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 1126–1135.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Qi Li, Zhenan Sun, Ran He, and Tieniu Tan. 2017. Deep supervised discrete hashing. *arXiv preprint arXiv:1705.10999*.

Genevieve Patterson, Chen Xu, Hang Su, and James Hays. 2014. The sun attribute database: Beyond categories for deeper scene understanding. *International Journal of Computer Vision*, 108(1-2):59–81.

Jake Snell, Kevin Swersky, and Richard S. Zemel. 2017. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 4077–4087.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. 2016. Matching networks for one shot learning. In *NeurIPS*, pages 3630–3638.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. 2011. The caltech-ucsd birds-200-2011 dataset.