CAD: Co-Adapting Discriminative Features for Improved Few-Shot Classification

Philip Chikontwe, Soopil Kim, Sang Hyun Park
Department of Robotics and Mechatronics, DGIST, South Korea
{philipchicco,soopilkim,shpark13135}@dgist.ac.kr

Abstract

Few-shot classification is a challenging problem that aims to learn a model that can adapt to unseen classes given a few labeled samples. Recent approaches pre-train a feature extractor, and then fine-tune for episodic meta-learning. Other methods leverage spatial features to learn pixel-level correspondence while jointly training a classifier. However, results using such approaches show marginal improvements. In this paper, inspired by the transformer style self-attention mechanism, we propose a strategy to cross-attend and re-weight discriminative features for few-shot classification. Given a base representation of support and query images after global pooling, we introduce a single shared module that projects features and cross-attends in two aspects: (i) query to support, and (ii) support to query. The module computes attention scores between features to produce an attention pooled representation of features in the same class that is later added to the original representation followed by a projection head. This effectively re-weights features in both aspects (i & ii) to produce features that better facilitate improved metric-based meta-learning. Extensive experiments on public benchmarks show our approach outperforms state-of-the-art methods by 3%~5%.

1. Introduction

Deep learning has achieved remarkable success in numerous computer vision tasks, reaching human-level performance in domains with sufficiently large-scale labeled training data. However, large-scale data is not always readily available due to costly curation and labeling. Hence, there is a ongoing research effort to design models that can learn to solve tasks using a limited number of labeled examples to alleviate this requirement. While several approaches such as transfer-, semi- and unsupervised learning have shown reasonable performance, learning to generalize with an extremely limited number of labeled samples is still
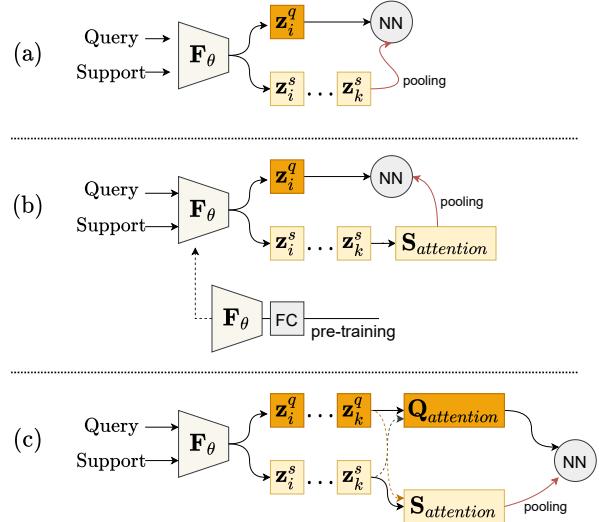

Figure 1. **Visual intuition** of the proposed framework compared to other approaches using attention. **(a)** Standard ProtoNet [43] architecture with nearest neighbour classifier (NN). **(b)** Adaptation of support embeddings only without using relations in the query sets, with backbone pre-training [54]. **(c)** Our method leverages attention from each set in a cross-relational perspective to improve classification performance by re-weighting features.

a challenge. Motivated by the ability of humans to rapidly adapt to new tasks using prior knowledge, the field of few-shot learning shows promise in achieving this goal.

Few-shot learning [22, 36, 43] aims to classify unlabeled query samples from unseen classes using only limited labeled support examples. Recent approaches have leveraged the meta-learning paradigm where transferable knowledge across a collection of tasks is learned and propagated to improve generalization. In particular, one of the promising approaches have used the meta-learning framework to optimize model parameters [11, 33, 36] with a few gradient steps, thus enabling neural network classifiers to quickly adapt to unseen classes. Other works employ similarity information between images (Fig. 1(a)), augmentation of

training data with generative techniques [15, 40], or two-stage approaches [7,35,42,53] that involve first pre-training the model on existing known classes and later use a meta-finetuning strategy. In the general meta-setup, tasks defined in the training phase mimic the testing phase to encourage generalization of models.

Though pre-training strategies show promising results, Chen *et al*. [7] argued that fine-tuning shows marginal improvements. Moreover, the standard technique of obtaining a base representation with global-average pooling is considered limited, since it is sensitive to object pose and discards key semantic details, and may be difficult to learn generalizable embeddings without being distracted by spurious features [18, 20]. Thus, recent works focus on learning finer distinctions using spatial features only via spatial attention mechanisms [19] or other forms of relational learning [20, 52] between query and support features to enhance performance. Nevertheless, methods leveraging this setting *i.e*., spatial feature-based learning, still report marginal improvements over purely discriminative approaches with the exception of some works that provide insight into the model reasoning procedure by visualizing object attended regions for interpretability.

Here, we argue that meta-learning with instance embeddings only is still viable and can improve few-shot performance when attending to the relevant features. For example, Ye *et al*. [54] attempt to address this and show that optimally discriminative features can be meta-learned using a set-to-set function based on the transformer model [47] (Fig. 1(b)). The self-attention mechanism in the Transformer presents several desirable properties for set based problems *i.e*., permutation invariance, interpolation, and contextualization. As a consequence, any meta-learner can leverage such a set-based transformation to adapt instance embeddings relative to other classes for improved prototype generation, or to produce better embeddings. However, while Ye *et al*. [54] showed improvements, embedding adaptation is performed on support samples only without leveraging query information, a key distinction with recent spatial techniques using cross-attention [19, 20, 52] to learn better correspondences. Moreover, the authors use a transfer-based approach requiring pre-training on base classes before meta-training.

In this work, we propose an end-to-end meta-learning strategy trained from scratch to co-attend to both support and query features using self-attention (Fig. 1(c)). This is useful to focus on common object features and avoid distractors for better matching. In particular, we introduce a single shared attention module that first projects global pooled base representations and computes attention scores per k-shot task via scaled-dot product. Given the scores per shot, we take the mean of the scores to obtain an attention pooled feature (dot product of features and scores) that is later added to the initial projected features before produc-

ing the final re-weighted features. Concretely, to improve support features *i.e*., *query* to *support*, attention scores between query and support features are employed to pool the initial support features producing a support prototype concatenated with the initial features, and vice versa for *support* to *query* to produce a query prototype that improves the initial query features. This strategy is conceptually equivalent to spatial based cross-attention that implicitly re-weights the base spatial maps to attend to relevant object regions. Finally, we employ a nearest neighbor classifier on the refined features for few-shot classification. The contributions of this paper can be summarized as follows:

- To improve model-based embedding adaptation, we propose to cross-attend support and query embeddings by re-weighting each instance relative to the other via self-attention.
- We reveal that implicitly re-weighting features with their prototypes (support/query) via self-attention improves metric based few-shot performance and adds minimal over-head in learnable parameters.
- Extensive evaluation of the proposed components verify the effectiveness of our method. Moreover, we show competitive results over state-of-the-art methods on several benchmark datasets.

2. Related works

Few-Shot Classification. Several works have been proposed to address few-shot learning [4,6,11,18,25,29,43,44, 48, 54, 56]. Existing methods broadly fall into the following categories: model initialization based methods [11, 12], metric learning methods [4,43,44,46,48], and hallucination based methods [14, 27, 31]. The first aims to learn a good model initialization by learning to finetune, where classifiers for unseen classes can be quickly adapted with a small number of gradient steps. However, these approaches have been reported to fail when large domain shifts exist between base and unseen classes [7].

On the other hand, metric learning methods address few-shot classification by learning to compare inputs by determining similarity during training. Here, predictions are often conditioned on distance metrics that include cosine similarity [48], euclidean distance to mean class representation [43], and relation modules [44]. Along this line of thought, recent approaches adopt a transfer-learning approach [7, 9, 14, 28, 30, 32, 39] where pre-training is employed with subsequent fine-tuning as a strong baseline for few-shot learning. Among these, our work falls under metric based approaches, and aims to improve transferability of features by co-adapting with a discriminative cross-attention module.
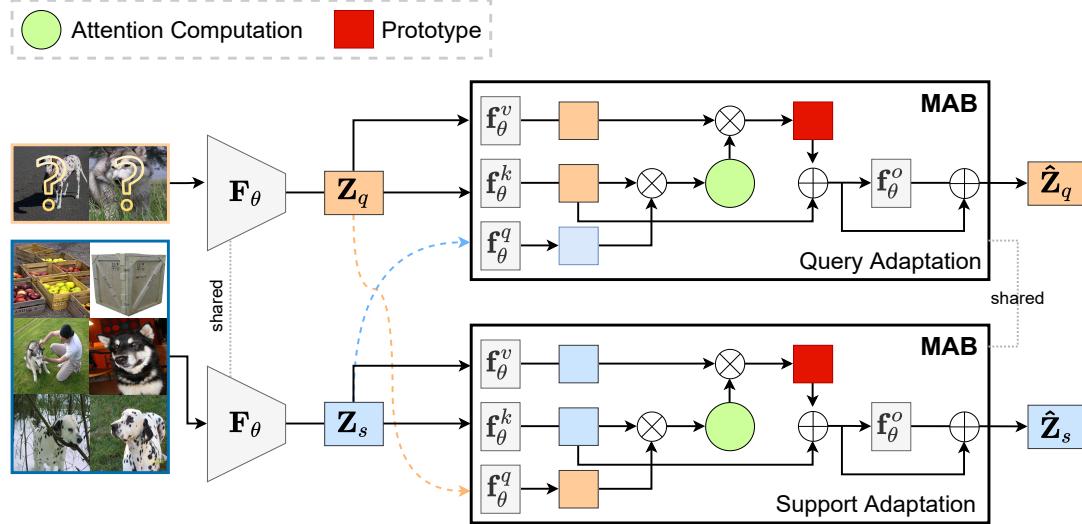
Figure 2. **Overview of the proposed approach**. We employ episodic meta-training from scratch to learn discriminative co-adapted embedding for support and query images. Concretely, a shared backbone extracts globally pooled features $\mathbf{Z}$ and performs feature adaptation using a shared self-attention module by considering support embeddings $\mathbf{Z_s}$ as keys and value pairs along with query $\mathbf{Z_q}$ to compute attention scores that re-weight $\mathbf{Z_s}$ to produce a prototype $\mathbf{p}$ added to $\mathbf{Z_s}$. Finally, we obtain $\hat{\mathbf{Z}}_\mathbf{s}$ after a projection via $\mathbf{f}_\theta^o$ (the same intuition is used for $\mathbf{Z_q}$ key/value pairs with $\mathbf{Z_s}$ as the query).

Few-shot Learning with Attention. Generally, attention can be employed to reveal the structural layout considering local patch neighborhoods [41] of an image, or highlight relevant features in set-based tasks by measuring similarities among inputs [2, 21, 23, 47]. This technique has been extensively explored in literature for both 2D and 3D visual tasks [34,50,58]. For example, CrossTransformer [10], CAN [19], and RENet [20] employ attention modules to project query and support features into the space of the other using spatial information. However, these methods incur several parameters over the backbone network and include additional objectives to regularize training.

Other settings such as incremental few-shot learning also use attention to regularize unseen class features by attending to seen classes [37]. FRN [52] formulate few-shot classification as a reconstruction problem following the work of Zhang *et al*. [55]. In the context of multi-domain few-shot learning, Liu *et al*. [29] use self-attention to select appropriate representations across domain specific backbones. Also, the benefit of using the Transformer [47] model is further explored in FEAT [54], where self-attention is used as a set-to-set transformation to adapt support representations. Inspired by prior works, we focus on the single domain setting, and propose a hybrid set-to-prototype cross-attention strategy that re-weights support set features by all query embeddings (or vice versa) to produce co-adapted features. Concretely, our approach is conceptually inductive, but can also be considered transductive given the use of unlabelled query samples similar to prior works [3, 57].

3. Method

In this section, we introduce the general problem setting for few-shot classification and its related formulations. Following, we describe the overall method as presented in Fig. 2, explain the attention mechanism and it's components, including the learning objective.

3.1. Problem Setting

Overview. We consider the standard few-shot learning problem in which we are given a labeled train set $\mathcal{D}_t$, an unlabeled query set $\mathcal{D}_q$, and a few labeled examples from a support set $\mathcal{S}$ sharing the same label space with the query set. The label space of the training and query set are non-overlapping *i.e.*, $\{Y_t \cap Y_q\} = \emptyset$, where $\{Y_t\}$ and $\{Y_q\}$ denote the training and query labels, respectively. Thus, given a set of labeled tasks from $\mathcal{D}_t$, and a few examples from the support set; the goal is to train a model that can generalize to novel tasks in the query set $\mathcal{D}_q$.

Towards this goal, meta-learning with episodic training can be employed. This scheme aims to improve generalization by mimicking the low-data setting encountered in inference by creating balanced episodes. An episode is formed by sampling disjoint data points from the support set consisting a few labeled points, and the query set where labels are used to calculate prediction errors per episode. Moreover, each episode defines an "N-way,k-shot" task, where N indicates the number of classes per episode, and k is the number of support examples per class.

Prototypical Baselines. In the context of Prototypical networks [43], one aims to construct an embedding space in which points cluster around a single *prototype* $\mathbf{p}_k \in \mathbb{R}^Z$ representation of each class k. Here, the objective is to learn an embedding function $\mathbf{F}_\theta(\mathbf{x}) : \mathbf{x} \in \mathbb{R}^{H \times W \times C} \to \mathbb{R}^Z$ to transform an input into a **z**-dimensional vector. To convey the description of the class as meta-data, the prototype of each class $\mathbf{p}_k$ is defined as the average of all embedded samples in that class:

$$\mathbf{p}_k = \frac{1}{|S_k|} \sum_{(\mathbf{x}_i, y_i) \sim S_k} \mathbf{F}_\theta(\mathbf{x}_i), \tag{1}$$

where S_k denotes samples from class k. To obtain the probability distribution over classes for a query sample $\mathbf{x}_q$, the softmax function on distances of the query to the prototypes is employed:

$$p_\theta(y_i = k | \mathbf{x}_q) = \frac{\exp(-d(\mathbf{F}_\theta(\mathbf{x}_q), \mathbf{p}_k))}{\sum_{k'} \exp(-d(\mathbf{F}_\theta(\mathbf{x}_q), \mathbf{p}_{k'}))}, \tag{2}$$

where $d(\cdot)$ is a distance function between query and the prototype. Consequently, $p_\theta(y_i = k | \mathbf{x}_q)$ is used to assign the correct class per sampled task, and the model learns to minimize the predictive error on query samples per task.

Though effective in practice, we argue that the prototype embeddings are not ideal to learn a discriminative model. Since every element in S_k is processed independently in the pooling operation (*i.e.* via mean), some information regarding interactions between them may be discarded making it difficult for the model to learn robust embeddings. In what follows, we describe how to co-adapt the embeddings via a transformer style set-to-prototype function using self-attention.

3.2. Instance Embedding Adaptation

In order to co-adapt support and query embeddings $\mathbf{x}_s$ and $\mathbf{x}_q$, we introduce an additional step where the initial embeddings are further transformed using self-attention. Here, attention is employed to find the relevant features to attend to in $\mathbf{x}_s$ and $\mathbf{x}_q$ by considering discriminative feature similarity only, and thus enables us to re-weight the features. Formally, let $z_q = \mathbf{F}_\theta(\mathbf{x}_q)$ and $z_s = \mathbf{F}_\theta(\mathbf{x}_s)$ be the initial query and support embeddings after the penultimate layer, each having vector dimension m *i.e.*, $z \in \mathbb{R}^m$. Our goal is to obtain adapted embeddings z'_q and z'_s.

Self-Attention. Following the work of Vaswani *et al.* [47] and its related extensions presented by Lee *et al.* [23] for set-based tasks, we employ an attention function $\varphi(\mathbf{Q}, \mathbf{K}, \mathbf{V})$ that measures similarity between a query vector $\mathbf{Q}$ with key-value pairs $\mathbf{K} \in \mathbb{R}^{n \times m}$ and $\mathbf{V} \in \mathbb{R}^{n \times m}$ via:

$$\varphi(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathrm{softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{m}})\mathbf{V}. \tag{3}$$

Concretely, the formulation introduced by Vaswani *et al.* [47] was extended to a multi-head attention block (MAB) where vectors $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ are first projected onto h different dimensional vectors. Here, $\varphi(\mathbf{Q}, \mathbf{K}, \mathbf{V})$ then becomes $\varphi(\mathbf{Q}W_j^{\mathbf{Q}}, \mathbf{K}W_j^{\mathbf{K}}, \mathbf{V}W_j^{\mathbf{V}})$, with each transformation having its own learnable parameters W. In this setting, each W can be modeled by functions $\mathbf{f}_\theta^q$, $\mathbf{f}_\theta^k$ and $\mathbf{f}_\theta^v$, respectively. Also, an additional function $\mathbf{f}_\theta^o$ takes the output of φ and adds the residual followed by a layer normalization operation. Mathematically,

$$\varphi^o = \varphi(\mathbf{Q}, \mathbf{K}, \mathbf{V}) + \mathbf{Q}, \tag{4}$$

$$\mathrm{MAB}_h(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \Phi(\varphi^o + \gamma(\mathbf{f}_\theta^o(\varphi^o))) \tag{5}$$

where γ is the ReLU function and φ^o is the output of applying attention on query key-value pairs after individual feature projection using functions $\mathbf{f}_\theta^{q,k,v}$ along with the concatenation of $\mathbf{Q}$. MAB_h defines a single attention block ($h = 1$) with an optional normalization layer Φ. It is worth noting that the $\mathbf{Q}$ notation used here is specifically related to features in the MAB_h module, and should be considered separate from the notion of queries in the few-shot setting.

**(a)**   **(b)**
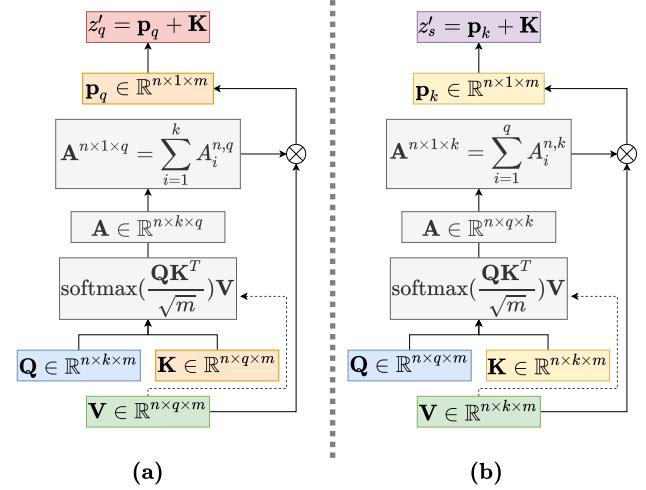
Figure 3. **Illustration of embedding adaptation**. **(a)** For query features, $\mathbf{Q} = z_s$ and $\mathbf{K} = \mathbf{V} = z_q$, where $\mathbf{p}_k$ and $\mathbf{p}_q$ are the support and query prototypes obtained via dot product of the average attention scores per-k shot with $\mathbf{V}$. **(b)**: Support features are modified by setting $\mathbf{Q} = z_q$ and $\mathbf{K} = \mathbf{V} = z_s$. Finally, n denotes the number of classes, with q the number of queries.

Cross-Adaptation with MAB. In the context of this work, to adapt features relative to the other, we consider two as-

pects: (i) *support-query*, and (ii) *query-support*, as presented in Fig. 3. For scenario (i), we employ the attention module as $\text{MAB}_h(z_s, z_q, z_q)$ (Fig. 3(a)), where support features z_s are considered as the query (**Q**), and the initial query embeddings z_q are key-value pairs. For (ii), the opposite is applied, *i.e.* $\text{MAB}_h(z_q, z_s, z_s)$ (Fig. 3(b)). Intuitively, this should produce embeddings that are weighted by attention reflecting feature similarity to obtain $z'_s = \text{MAB}_h(z_q, z_s, z_s)$, and $z'_q = \text{MAB}_h(z_s, z_q, z_q)$, respectively.

To this end, we wish to note that Lee *et al.* [23] employ Eq. 4 & Eq. 5 by making **Q** a learnable parameter in order to perform attention-based pooling in set-based tasks *i.e.* ($\mathbf{K} \in \mathbb{R}^{n \times k \times m}$, $\mathbf{V} \in \mathbb{R}^{n \times k \times m}$) are a set of features and $\mathbf{Q} = W \in \mathbb{R}^{n \times m}$ is a tensor initialized from a normal distribution to produce a pooled tensor $\mathbf{O} \in \mathbb{R}^{n \times m}$, with n as the batch dimension, and k the number of samples in the set, respectively. This approach has been shown to produce better pooled embeddings over simply employing the mean directly. In this work, employing Eq. 4 reveals a tensor shape mismatch since few-shot inputs consist q-queries and k-shot samples for the support set in a meta-setup. To address this, we modify Eq. 4 and Eq. 5 as:

$$\hat{\varphi}^o = \varphi(\mathbf{Q}, \mathbf{K}, \mathbf{V}) + \mathbf{K}, \quad (6)$$

$$\text{MAB}_h(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \Phi(\hat{\varphi}^o + \gamma(\mathbf{f}_\theta^o(\hat{\varphi}^o))). \quad (7)$$

In addition, note that $\varphi(\cdot)$ will result in different attention scores for *query-support* and *support-query i.e.*, $\mathbf{A}_s \in \mathbb{R}^{n \times q \times k}$ and $\mathbf{A}_q \in \mathbb{R}^{n \times k \times q}$ for k-shot with q query examples. Thus, in each setting we take the average over k or q to obtain the final scores. For example, in support adaptation, this implies we obtain $\mathbf{A}_s \in \mathbb{R}^{n \times 1 \times k}$ and taking the dot product with $\mathbf{V}$ produces a prototype embedding, vice versa for queries. The support/query prototypes reflect the relevant features later added to the initial $\mathbf{K}$ to obtain the final embeddings. Consequently, the prototype of the adapted support embeddings are now obtained via:

$$\hat{\mathbf{p}}_k = \frac{1}{|S_k|} \sum_{(\mathbf{x}_i, y_i) \sim S_k} z'_s, \quad (8)$$

and the distribution over classes is:

$$p_\theta(y_i = k | \mathbf{x}_q) = \frac{\exp(-d(z'_q, \hat{\mathbf{p}}_k))}{\sum_{k'} \exp(-d(z'_q, \hat{\mathbf{p}}_{k'}))}. \quad (9)$$

To optimize the backbone and proposed modules, we train the model following the standard setting proposed in Prototypical networks (inductive only) using a cross-entropy loss on the negative distances to minimize the query prediction error per task. During inference, all modules are used as is, maintaining the design of the training phase without omission of any components.

4. Experiments

In this section, details on the datasets employed along with implementation settings are presented. We evaluate our proposed method on standard few-shot benchmarks and compare results with recent state-of-the-art methods.

Datasets. To verify the proposed method, we address few shot classification under two scenarios: generic object recognition and fine-grained image classification. For this purpose, we employ the *mini*ImageNet [8, 36], *tiered*ImageNet [38], CIFAR-FS [5], and CUB-200-2011 [49] datasets.

***mini*ImageNet** consists of a subset of 100 object classes from ImageNet [8] with 600 images per class. We follow the setting proposed by Ravi *et al.* [36] to randomly select 64 base, 16 validation, and 20 novel classes, respectively. ***tiered*ImageNet** is a larger subset of ImageNet with 351/97/160 sub-classes for training/validation/testing stemming from 20/6/8 super-classes. It is a more challenging dataset as the splits are disjoint in terms of super-classes and requires better generalization. **CUB-200-2011** consists of 200 classes with a total of $11,788$ images. Following the protocols of Hilliard *et al.* [17], the dataset is randomly split into 100 base, 50 validation, and 50 novel classes. **CIFAR-FS** is built upon CIFAR-100 and has 100 classes, each containing 600 images. We use the same split of 64/16/20 as in Bertinetto *et al.* [5].

Implementation Details. Following recent works [19, 20, 55], we adopt ResNet12 [16] as the backbone network. The backbone take images of size 84×84 as input and produces embeddings $z_{q,s} \in \mathbb{R}^{640}$ after global pooling. We train our model in the '5-way 1-shot' and '5-way 5-shot' setting with standard normalization and augmentation techniques as in prior work [20]. Moreover, our attention block is shared between adaptation steps and only uses a single head (h=1) with dimensions of the fully-connected layers set to 640, and $d(\cdot)$ is the euclidean function. In training, Adam optimizer was used with a learning rate of 0.003 without decay or scheduling. 1-shot models were trained for 300 epochs with 200 tasks sampled per epoch, resulting in $60,000$ tasks. On the other hand, 5-shot models were trained for 200 epochs totaling $40,000$ tasks similar to Chen *et al.* [7]. During evaluation, we meta-test 15 query samples per class in each episode and report the average accuracy with 95% confidence interval over $2,000$ randomly sampled test episodes.

Comparison with State-of-the-Art Methods. Tables 1 and 2 compare the proposed method with current state-of-the-art few-shot methods [19, 20, 52, 54, 55]. Our approach shows consistent improvements over several methods in all evaluated settings. Notably, we observe sig-

Method	Backbone	miniImageNet		tieredImageNet	
		1−shot	5−shot	1−shot	5−shot
ProtoNet [43]	ResNet12	62.39±0.21	80.53±0.14	68.23±0.23	84.03±0.16
MetaOptNet [24]	ResNet12	62.64±0.82	78.63±0.46	65.99±0.72	81.56±0.53
SimpleShot [51]	ResNet18	62.85±0.20	80.02±0.14	-	-
MatchNet [48]	ResNet12	63.08±0.80	75.99±0.60	68.50±0.92	80.60±0.71
RFS-simple [45]	ResNet12	62.02±0.63	79.64±0.44	69.74±0.72	84.41±0.55
S2M2 [32]	ResNet34	63.74±0.18	79.45±0.12	-	-
NegMargin [28]	ResNet12	63.85±0.81	81.57±0.56	-	-
CTM [26]	ResNet18	64.12±0.82	80.51±0.13	68.41±0.39	84.28±1.73
CAN [19]	ResNet12	63.85±0.48	79.44±0.34	69.89±0.51	84.23±0.37
DeepEMD [55]	ResNet12	65.91±0.82	82.41±0.56	71.16±0.87	86.03±0.58
FEAT [54]	ResNet12	66.78±0.20	82.05±0.14	70.80±0.23	84.79±0.16
RENet [20]	ResNet12	67.60±0.44	82.58±0.30	71.61±0.51	85.28±0.35
FRN [52]	ResNet12	66.45±0.19	82.83±0.13	72.06±0.22	86.89±0.14
EPNet [39]	ResNet12	66.50±0.89	81.06±0.61	76.53±0.87	87.32±0.64
Dist-Calib [53]	WRN28	68.51±0.55	82.88±0.42	78.19±0.25	89.90±0.41
SLK-MS [57]	ResNet18	73.10	82.82	**79.99**	86.55
EPNet [39]	WRN28	70.74±0.85	84.34±0.53	78.50±0.91	88.36±0.57
SLK-MS [57]	WRN28	75.17	84.28	81.13	87.69
Ours	ResNet12	**77.56±0.72**	**87.68±0.57**	77.55±0.74	**90.73±0.54**

Table 1. Few-shot classification accuracy on *mini*ImageNet and *tiered*ImageNet in the 5-way k-shot setting (mean accuracy in a ±95% confidence interval is reported)

Method	Backbone	1−shot	5−shot
Cosine [7]	ResNet34	68.00±0.83	84.50±0.51
MatchNet [48]	ResNet12	71.87±0.85	85.08±0.57
NegMargin [28]	ResNet18	72.66±0.85	89.40±0.43
S2M2 [32]	ResNet18	71.81±0.43	86.22±0.53
S2M2 [32]	ResNet34	72.92±0.83	86.55±0.51
FEAT* [54]	ResNet12	73.27±0.22	85.77±0.14
DeepEMD [55]	ResNet12	75.65±0.83	88.69±0.50
ProtoNet [43]	ResNet12	66.09±0.92	82.50±0.58
RENet [20]	ResNet12	79.49±0.44	91.11±0.24
SLK-MS [57]	ResNet18	81.88	88.55
EPNet [39]	ResNet12	82.85±0.81	91.32±0.41
FRN [52]	ResNet12	**83.55±0.19**	**92.92±0.10**
Ours	ResNet12	82.95±0.67	90.80±0.51

(a) Results on CUB-200-2011 dataset.

Method	Backbone	1−shot	5−shot
Cosine [7]	ResNet34	60.39±0.28	72.85±0.65
S2M2 [32]	ResNet18	63.66±0.17	76.07±0.19
S2M2 [32]	ResNet34	62.77±0.23	75.75±0.13
ProtoNet [43]	ResNet12	72.20±0.70	83.50±0.50
MetaOptNet [24]	ResNet12	72.80±0.70	85.00±0.50
Boosting [13]	WRN28	73.60±0.30	86.00±0.20
RENet [20]	ResNet12	74.51±0.46	86.60±0.32
Ours	ResNet12	**79.97±0.72**	**94.13±0.41**

(b) Results on CIFAR-FS dataset.

Table 2. Few-shot classification accuracy on CUB-200-2011 and CIFAR-FS in the 5-way k-shot setting (mean accuracy in the ±95% confidence interval). "*" : denotes results reproduced by [20].

nificant gains in both 1- and 5-shot settings, highlighting the effectiveness of our approach despite its simplicity. For example, on *mini*ImageNet, our approach shows +3% and +4% gain in 1-shot and 5-shot settings over the best method. We observed similar results on the challenging *tiered*ImageNet with +1% gains, except for CUB where our approach shows results on par with current state-of-the-art methods. While closely related method RENet [20] employs spatial relational learning of query/support features, our technique solely benefits from discriminative co-adaptation to enhance performance. Recent work FRN [52] equally shows competitve performance, however FRN reformulates few-shot classification to leverage reconstruction of related features, and requires a two-stage process with pre-training followed by episodic fine-tuning. In contrast, our approach can be trained end-to-end from scratch with episodic training.

As shown in Table 2, our method is highly competitive to the closely related method FEAT [54]. Note that FEAT is another transfer learning method leveraging self-attention on support samples only, and uses extra regularization terms in the episodic training step. Herein, the results validate our initial hypothesis that discriminative learning alone can still show gains over existing baselines. On the other hand, while our approach does not report the best result on CUB-200, we are on par with the best method FRN and produce the best performance on CIFAR-FS ($\approx 5\%$). In addition, while approaches such as Boosting [13] use self-supervision and auxiliary losses with extra unlabeled samples for semi-supervised few-shot learning, we show that a smaller backbone is still competitive.

5. Ablation Study

In this section, we further validate the proposed method by analyzing the effect of omitting certain components/modules on model performance, and also investigate the challenging cross-domain generalization setting. Moreover, we provide quantitative evaluation on the benefit of the proposed attention strategy in a Prototypical baseline.

5.1. Improved Prototypical Baseline

Model	Backbone	1-shot	5-shot
ProtoNet* [43]	ResNet12	51.61±0.44	72.28±0.36
ProtoNet** w/ Eq. 6	ResNet12	**59.57±0.61**	**84.29±0.40**

Table 3. Performance comparison of standard ProtoNet baseline and a ProtoNet using self-attention on *mini*ImageNet dataset in the 5-way k-shot scenario. "*": denotes reproduced results and "**" denotes a model using a non-parametric version of the proposed method following Eq. 6.

Table 3 summarizes the benefit of employing the proposed cross adaptation of embeddings in ProtoNet [43]. We re-train the ProtoNet model with a non-parametric version of the proposed method *i.e.*, no use of fully-connected layers $\mathbf{f}_\theta^q$, $\mathbf{f}_\theta^k$, $\mathbf{f}_\theta^v$ and $\mathbf{f}_\theta^o$. In particular, given the base representation after global average pooling, we directly apply Eq. 6 for both support-query and query-support adaptation, without ReLU or normalization layers. Results show that simply re-weighting features is beneficial, especially in the 5-shot setting where larger gains were observed.

5.2. Cross-Domain Few-Shot Classification

Model	Backbone	1-shot	5-shot
ProtoNet* [43]	ResNet18	-	62.02±0.70
SimpleShot* [51]	ResNet18	48.56	65.63
MatchNet* [48]	ResNet10	36.61±0.53	55.23±0.83
NegMargin* [28]	ResNet18	-	69.30±0.70
MetaOptNet* [24]	ResNet12	44.79±0.75	64.98±0.68
ASL [1]	ResNet18	46.85±0.75	70.37±1.02
FRN [52]	ResNet12	54.11±0.19	77.09±0.15
Ours	ResNet12	**74.10±0.75**	**86.37±0.60**

Table 4. Performance comparison in the cross-domain setting: *mini*ImageNet → CUB-200-2011 in the 5-way k-shot scenario. "*" : denotes results reported by [52].

In Table 4, we evaluate on the challenging cross-domain setting of *mini*ImageNet to CUB. Following the splits introduced by Chen *et al.* [7], we trained our model on *mini*ImageNet only {base+val+test}, and meta-test on CUB test split. We report significant improvements over related methods especially in the 1-shot setting. Our intuition is

that in the cross-domain setup, cross adaptation of features leads to better generalization by avoiding distracting features via attention.

5.3. Effect of the Self-Attending Modules

MAB_h^S	MAB_h^Q	MAB_h^*	1-shot	5-shot
✓	✗	✗	56.00±0.79	85.10±0.61
✗	✓	✗	76.33±0.73	80.10±0.74
✗	✗	✓	56.01±0.86	79.94±0.72
✓	✓	✗	**77.56±0.72**	**87.68±0.57**

Table 5. Evaluation on the contribution of attention modules in the proposed method with ResNet12 on *mini*ImageNet. MAB_h^* denotes using self-attention only on both query and support features i.e., $z_s' = \mathrm{MAB}_h(z_s, z_s, z_s)$ & $z_q' = \mathrm{MAB}_h(z_q, z_q, z_q)$. MAB_h^S implies support adaptation: $z_s' = \mathrm{MAB}_h(z_q, z_s, z_s)$, whereas MAB_h^Q is query adaptation $z_q' = \mathrm{MAB}_h(z_s, z_q, z_q)$, respectively.

Here, we evaluate the benefit of co-adapting features in a cross-relational way with the shared attention module MAB_h. Table 5 shows the mean accuracy of decomposing our proposed method to assess the effect of omitting or including one of the components. When self-attention is employed on support samples only *i.e.*, MAB_h^S, performance drops significantly, especially in the 1-shot setting where results are below existing state-of-the-art baselines as opposed to a higher 5-shot accuracy that may imply sensitivity to the number of samples for prototype generation. Interestingly, adaptation of queries (MAB_h^Q) alone shows consistent results, yet is marginally lower than the proposed method in the 5-shot setting.

Finally, we assess the benefit of using self-attention only similar to FEAT [54]. However, our implementation applies support-support and query-query (MAB_h^*) only without any adaptation, nor do we employ regularization terms in the learning objective. Our findings suggest that self-attention alone does not improve over the baseline models such as ProtoNet (Table 1), and also suffers in the extreme 1-shot setting.

5.4. Qualitative Results

The effects of cross-adaptation of features via the attention scores per branch are shown in Fig. 4. We randomly sampled a task on *mini*ImageNet and varied the number of query images in the 5-way 5-shot setting *i.e.*, $\{1, 2, 3\}$ queries with 5 support images, respectively. For (a), the attention score of a single query will have no effect in adaptation, given its a single sample. We observed that our method adds more weight to the most similar support samples, and lowered weights for samples with extremely varied background or color. Moreover in (b), due to the presence of multiple objects at a distance (support image w/ 0.08), less

| 1.00 | 0.21 | 0.18 | **0.35** | 0.09 | 0.17 |

(a)

| **0.74** | 0.26 | 0.16 | **0.26** | 0.25 | 0.25 | 0.08 |

(b)

| **0.37** | 0.37 | 0.26 | 0.11 | 0.16 | 0.26 | **0.26** | 0.21 |

(c)
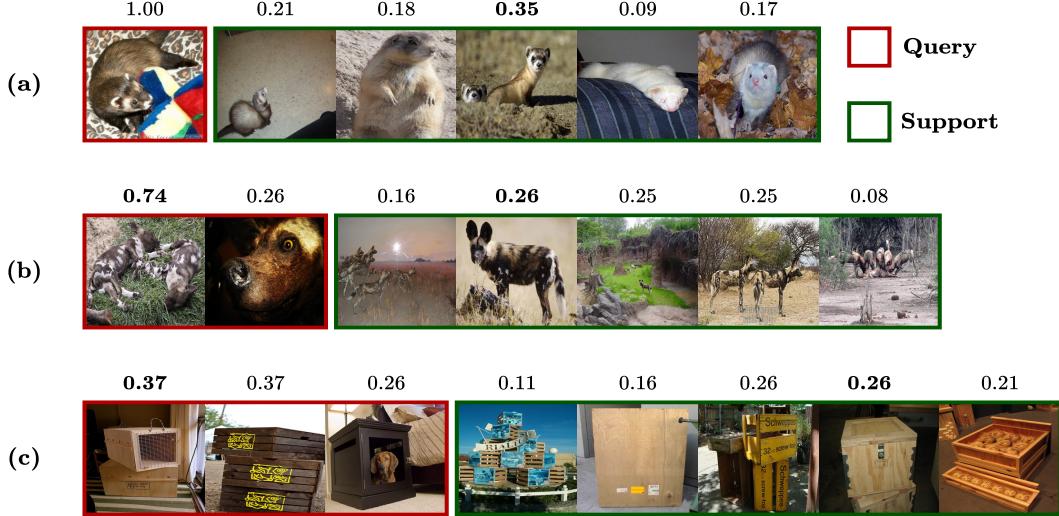
Figure 4. **Illustration of the attention mechanism**. **(a)** Attention scores of our 5-way 5-shot model on the support set using a single query image on a randomly sampled task *mini*ImageNet testing split. **(b)** Scores for a task with 2-query images each re-weighted relative to the support samples, and **(c)** shows scores using 3-query images.

| (a) | (b) | (c) | (d) |
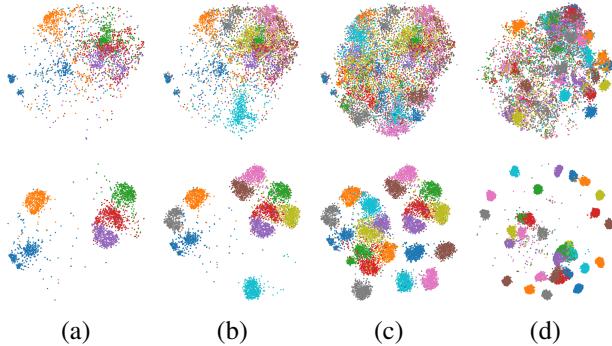
Figure 5. **TSNE of features**. The top row shows base features after global pooling only whereas the bottom shows features after adaptation on *mini*ImageNet w/ a 5-way 5-shot model. **(a)**, **(b)** and **(c)** are embeddings of 5, 10 and 20 classes in the test-set. Finally, **(d)** are embeddings of 30 random classes in the train-set.

weight was given. Interestingly, the model penalizes the query with a close-up of the *wild dog*. We observed similar trends as query images increased, as in the case (c); querys with *boxes* only were weighted equally, whereas the query with the dog was penalized. Given that our method does not explicity employ visual correspondence to re-weights features, this shows the viability and robustness of our method.

In Fig. 5, we present a visualization of embeddings before and after adaptation. Generally, few-shot models trained episodically can not form accurate clusters of the existing classes due to limited samples in meta-training, unless explicitly enforced. We observed that after applying our strategy on base representations, features were more compact and embeddings were clustered relative to the true class. We owe this to our method adding attention pooled prototypes to the initial features, ensuring either query or support samples are more robust.

Limitations. Our method shows reasonable improvements over recent approaches leveraging instance embeddings only. However, producing interpretable maps that explicitly highlight pixel-level correspondence between images rather than attention scores may be more desirable. Moreover, it may be computationally prohibitive when larger backbones are employed with more heads. Thus, extension of our method to fully-convolutional MAB with spatial features may be interesting, and is left for future research.

6. Conclusion

In this work, we revisited discriminative self-attention for few-shot classification via cross-relational adaptation of support and queries. Our work has shown that attending to features in a single view only may be limited, instead cross-adaptation that attends to relevant features via re-weighting with task prototypes can boost performance and generalize better in the cross-domain setting. We also demonstrate state-of-the-art performance without the requirement of transfer learning.

References

[1] Arman Afrasiyabi, Jean-François Lalonde, and Christian Gagné. Associative alignment for few-shot image classification. In *ECCV*, pages 18–35. Springer, 2020. 7

[2] Sion An, Soopil Kim, Philip Chikontwe, and Sang Hyun Park. Few-shot relation learning with attention for eeg-based motor imagery classification. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10933–10938. IEEE. 3

[3] Peyman Bateni, Jarred Barber, Jan-Willem van de Meent, and Frank Wood. Enhancing few-shot image classification with unlabelled examples. In *IEEE Winter. Conf. Application. Comput. Vis.*, pages 2796–2805, 2022. 3

[4] Peyman Bateni, Raghav Goyal, Vaden Masrani, Frank Wood, and Leonid Sigal. Improved few-shot visual classification. In *CVPR*, pages 14493–14502, 2020. 2

[5] Luca Bertinetto, Joao F Henriques, Philip Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. In *ICLR*, 2018. 5

[6] Da Chen, Yuefeng Chen, Yuhong Li, Feng Mao, Yuan He, and Hui Xue. Self-supervised learning for few-shot image classification. In *ICASSP*, pages 1745–1749. IEEE, 2021. 2

[7] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. In *ICLR*, 2019. 2, 5, 6, 7

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255. Ieee, 2009. 5

[9] Guneet Singh Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. A baseline for few-shot image classification. In *ICLR*, 2019. 2

[10] Carl Doersch, Ankush Gupta, and Andrew Zisserman. Crosstransformers: spatially-aware few-shot transfer. In *NeurIPS*, 2020. 3

[11] Chelsea Finn, P. Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, 2017. 1, 2

[12] Chelsea Finn, Kelvin Xu, and Sergey Levine. Probabilistic model-agnostic meta-learning. In *NeurIPS*, 2018. 2

[13] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Boosting few-shot visual learning with self-supervision. In *ICCV*, pages 8059–8068, 2019. 6

[14] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, pages 4367–4375, 2018. 2

[15] Bharath Hariharan and Ross B. Girshick. Low-shot visual recognition by shrinking and hallucinating features. *ICCV*, pages 3037–3046, 2017. 2

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 5

[17] Nathan Hilliard, Lawrence Phillips, Scott Howland, Artëm Yankov, Courtney D Corley, and Nathan O Hodas. Few-shot learning with metric-agnostic conditional embeddings, 2018. 5

[18] Jie Hong, Pengfei Fang, Weihao Li, Tong Zhang, Christian Simon, Mehrtash Harandi, and Lars Petersson. Reinforced attention for few-shot learning and beyond. In *CVPR*, pages 913–923, 2021. 2

[19] Ruibing Hou, Hong Chang, Bingpeng Ma, S. Shan, and Xilin Chen. Cross attention network for few-shot classification. In *NeurIPS*, 2019. 2, 3, 5, 6

[20] Dahyun Kang, Heeseung Kwon, Juhong Min, and Minsu Cho. Relational embedding for few-shot classification. In *ICCV*, pages 8822–8833, 2021. 2, 3, 5, 6

[21] Soopil Kim, Sion An, Philip Chikontwe, and Sang Hyun Park. Bidirectional rnn-based few shot learning for 3d medical image segmentation. In *AAAI*, volume 35, pages 1808–1816, 2021. 3

[22] Gregory Koch, Richard Zemel, Ruslan Salakhutdinov, et al. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille, 2015. 1

[23] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International Conference on Machine Learning*, pages 3744–3753. PMLR, 2019. 3, 4, 5

[24] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. *CVPR*, pages 10649–10657, 2019. 6, 7

[25] Aoxue Li, Tiange Luo, Tao Xiang, Weiran Huang, and Liwei Wang. Few-shot learning with global class representations. In *ICCV*, pages 9715–9724, 2019. 2

[26] Hongyang Li, David Eigen, Samuel Dodge, Matthew Zeiler, and Xiaogang Wang. Finding task-relevant features for few-shot learning by category traversal. In *CVPR*, pages 1–10, 2019. 6

[27] Kai Li, Yulun Zhang, Kunpeng Li, and Yun Fu. Adversarial feature hallucination networks for few-shot learning. In *CVPR*, pages 13470–13479, 2020. 2

[28] Bin Liu, Yue Cao, Yutong Lin, Qi Li, Zheng Zhang, Mingsheng Long, and Han Hu. Negative margin matters: Understanding margin in few-shot classification. In *ECCV*, pages 438–455. Springer, 2020. 2, 6, 7

[29] Lu Liu, William L. Hamilton, Guodong Long, Jing Jiang, and Hugo Larochelle. A universal representation transformer layer for few-shot image classification. In *ICLR*, 2021. 2, 3

[30] Yanbin Liu, Juho Lee, Minseop Park, Saehoon Kim, Eunho Yang, Sung Ju Hwang, and Yi Yang. Learning to propagate labels: Transductive propagation network for few-shot learning. In *ICLR*, 2018. 2

[31] Qinxuan Luo, Lingfeng Wang, Jingguo Lv, Shiming Xiang, and Chunhong Pan. Few-shot learning via feature hallucination with variational inference. In *IEEE Winter. Conf. Application. Comput. Vis.*, pages 3963–3972, 2021. 2

[32] Puneet Mangla, Nupur Kumari, Abhishek Sinha, Mayank Singh, Balaji Krishnamurthy, and Vineeth N Balasubramanian. Charting the right manifold: Manifold mixup for few-shot learning. In *IEEE Winter. Conf. Application. Comput. Vis.*, pages 2218–2227, 2020. 2, 6

[33] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms, 2018. 1

[34] Shi Qiu, Saeed Anwar, and Nick Barnes. Geometric back-projection network for point cloud classification. *IEEE Transactions on Multimedia*, 2021. 3

[35] Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. Rapid learning or feature reuse? towards understanding the effectiveness of maml. In *ICLR*, 2019. 2

[36] S. Ravi and H. Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017. 1, 5

[37] Mengye Ren, Renjie Liao, Ethan Fetaya, and R. Zemel. Incremental few-shot learning with attention attractor networks. In *NeurIPS*, 2019. 3

[38] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. In *ICLR*, 2018. 5

[39] Pau Rodríguez, Issam Laradji, Alexandre Drouin, and Alexandre Lacoste. Embedding propagation: Smoother manifold for few-shot classification. In *ECCV*, pages 121–138. Springer, 2020. 2, 6

[40] Eli Schwartz, Leonid Karlinsky, J. Shtok, Sivan Harary, Mattias Marder, Abhishek Kumar, R. Feris, R. Giryes, and A. Bronstein. Delta-encoder: an effective sample synthesis method for few-shot object recognition. In *NeurIPS*, 2018. 2

[41] Eli Shechtman and Michal Irani. Matching local self-similarities across images and videos. In *CVPR*, pages 1–8. IEEE, 2007. 3

[42] Zhiqiang Shen, Zechun Liu, Jie Qin, Marios Savvides, and Kwang-Ting Cheng. Partial is better than all: Revisiting fine-tuning strategy for few-shot learning. In *AAAI*, volume 35, pages 9594–9602, 2021. 2

[43] J. Snell, Kevin Swersky, and R. Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, 2017. 1, 2, 4, 6, 7

[44] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *CVPR*, pages 1199–1208, 2018. 2

[45] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: a good embedding is all you need? In *ECCV*, pages 266–282. Springer, 2020. 6

[46] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, et al. Meta-dataset: A dataset of datasets for learning to learn from few examples. In *ICLR*, 2019. 2

[47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, pages 5998–6008, 2017. 2, 3, 4

[48] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *NeurIPS*, 29:3630–3638, 2016. 2, 6, 7

[49] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 5

[50] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, pages 7794–7803, 2018. 3

[51] Yan Wang, Wei-Lun Chao, Kilian Q. Weinberger, and L. V. D. Maaten. Simpleshot: Revisiting nearest-neighbor classification for few-shot learning, 2019. 6, 7

[52] Davis Wertheimer, Luming Tang, and Bharath Hariharan. Few-shot classification with feature map reconstruction networks. In *CVPR*, pages 8012–8021, 2021. 2, 3, 5, 6, 7

[53] Shuo Yang, Lu Liu, and Min Xu. Free lunch for few-shot learning: Distribution calibration. In *ICLR*, 2020. 2, 6

[54] Han-Jia Ye, Hexiang Hu, De-Chuan Zhan, and Fei Sha. Few-shot learning via embedding adaptation with set-to-set functions. In *CVPR*, pages 8808–8817, 2020. 1, 2, 3, 5, 6, 7

[55] Chi Zhang, Yujun Cai, Guosheng Lin, and Chunhua Shen. Deepemd: Few-shot image classification with differentiable earth mover's distance and structured classifiers. in 2020 ieee. In *CVPR*, pages 12200–12210, 2020. 3, 5, 6

[56] Jian Zhang, Chenglong Zhao, Bingbing Ni, Minghao Xu, and Xiaokang Yang. Variational few-shot learning. In *ICCV*, pages 1685–1694, 2019. 2

[57] Imtiaz Masud Ziko, Malik Boudiaf, Jose Dolz, Eric Granger, and Ismail Ben Ayed. Transductive few-shot learning: Clustering is all you need?, 2021. 3, 6

[58] Daniel Zoran, Mike Chrzanowski, Po-Sen Huang, Sven Gowal, Alex Mott, and Pushmeet Kohli. Towards robust image classification using sequential attention models. In *CVPR*, pages 9483–9492, 2020. 3