The More, The Better? Active Silencing of Non-Positive Transfer for Efficient Multi-Domain Few-Shot Classification

Xingxing Zhang*
Dept. of Comp. Sci. & Tech., Institute for AI, BNRist Center, THBI Lab, Tsinghua University
Beijing, China
xxzhang2020@mails.tsinghua.edu.cn

Zhizhe Liu*
Institute of Information Science, Beijing Jiaotong University
Beijing, China
zhzliu@bjtu.edu.cn

Weikai Yang
School of Software, BNRist Center, Tsinghua University
Beijing, China
vicayang496@gmail.com

Liyuan Wang
Dept. of Comp. Sci. & Tech., Institute for AI, BNRist Center, THBI Lab, Tsinghua University
Beijing, China
wly19@mails.tsinghua.edu.cn

Jun Zhu†
[1]Dept. of Comp. Sci. & Tech., Institute for AI, BNRist Center, THBI Lab, Tsinghua University, Beijing, China
[2]Peng Cheng Laboratory, Pazhou Laboratory (Huangpu)
Guangzhou, China
dcszj@tsinghua.edu.cn

ABSTRACT

Few-shot classification refers to recognizing several novel classes given only a few labeled samples. Many recent methods try to gain an adaptation benefit by learning prior knowledge from more base training domains, aka. multi-domain few-shot classification. However, with extensive empirical evidence, we find more is not always better: current models do not necessarily benefit from pre-training on more base classes and domains, since the pre-trained knowledge might be non-positive for a downstream task. In this work, we hypothesize that such redundant pre-training can be avoided without compromising the downstream performance. Inspired by the selective activating/silencing mechanism in the biological memory system, which enables the brain to learn a new concept from a few experiences both quickly and accurately, we propose to actively silence those redundant base classes and domains for efficient multi-domain few-shot classification. Then, a novel data-driven approach named Active Silencing with hierarchical Subset Selection (AS3) is developed to address two problems: 1) finding a subset of base classes that adequately represent novel classes for efficient positive transfer; and 2) finding a subset of base learners (*i.e.*, domains) with confident accurate prediction in a new domain. Both problems are formulated as distance-based sparse subset selection. We extensively evaluate AS3 on the recent META-DATASET benchmark as well as MNIST, CIFAR10, and CIFAR100, where AS3 achieves over 100% acceleration while maintaining or even improving accuracy. Our code and Appendix are available at https://github.com/indussky8/AS3.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning algorithms**; *Image representations.*

KEYWORDS

multi-domain, pre-trained knowledge, subset selection, efficient few-shot learning

ACM Reference Format:
Xingxing Zhang, Zhizhe Liu, Weikai Yang, Liyuan Wang, and Jun Zhu. 2022. The More, The Better? Active Silencing of Non-Positive Transfer for Efficient Multi-Domain Few-Shot Classification. In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisboa, Portugal.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3503161.3548349

1 INTRODUCTION

Learning a new task from a few annotated samples, *i.e.*, few-shot learning [11, 18, 21, 28, 49], remains a great challenge for machine learning systems. It especially shows a noticeable gap compared to the ability of humans to quickly understand new concepts from just one or a handful of examples [22]. A promising direction to address this challenge is developing methods that are capable of performing transfer learning across the collective data of many preexisting tasks [13, 16, 43]. As a result, many *multi-domain few-shot classification* methods are proposed recently [4, 9, 25, 41]. Fig. 1 presents a general framework of this kind of methods. During the first learning stage, a pre-trained model is built using a number of large labeled datasets corresponding to different domains. While in the second stage, the pre-trained model is adapted to several novel classes of a new domain given only a few labeled samples.

*Equal contribution.
†Corresponding author.

MM '22, October 10–14, 2022, Lisboa, Portugal

ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3548349

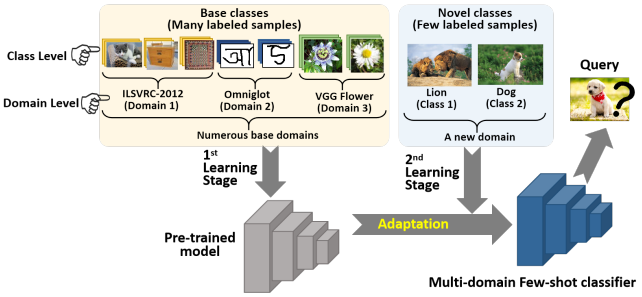


Figure 1: A general framework for multi-domain few-shot classification: Given a number of labeled datasets, we first learn a pre-trained model; and then adapt it to a new domain with a few labeled samples from several novel classes.

One useful yet seldom investigated question in Fig. 1 is *whether it is better for adaptation by employing more base classes and domains.* With extensive empirical evidence (see Sec. 5.2), we find more is not always better. This is reasonable since without access to enough labeled samples in a new domain, knowledge transfer from numerous base classes (*resp.*, base domains) to novel classes (*resp.*, a new domain) is difficult to determine, which might be positive, negative, or neither. If the transfer is negative, *i.e.*, learning a base class (*resp.*, domain) in the first stage (Fig. 1) makes the prediction of a new class (*resp.*, domain) worse, then employing such a class (*resp.*, domain) will jeopardize the performance of few-shot classifiers. In addition, when employing more base classes (*resp.*, domains), the first learning stage will naturally require more computation overhead, which is also a core problem under addressed as claimed in [9]. Thus, a successful approach for multi-domain few-shot classification should 1) not only address the regular challenge of few-shot classification, *i.e.*, how to adapt a model learned from base classes to several novel classes; 2) but also discover what knowledge in base classes and domains should be employed/silenced, to achieve generalization efficiently with strong positive transfer.

By contrast, the biological memory system can learn new experiences *both quickly and accurately* by choosing which cells in a given brain region are active/silent at memory encoding and retrieval. An emerging concept is that a given memory is supported by an engram complex, composed of functionally connected engram cell ensembles dispersed across multiple brain regions, with each ensemble supporting a component of the overall memory [19]. As shown in Fig. 2(a), retrieval of a target memory may lead to forgetting of currently irrelevant competing memories, as well as irrelevant engram ensembles in relevant memories. Understanding how the brain encodes, stores, and uses information for new experiences, especially at the level of the engram, we derive a novel, simple yet effective approach named Active Silencing with hierarchical Subset Selection (AS3) for efficient multi-domain few-shot classification. In particular, to solve the two key issues above, we try to enforce an efficient positive knowledge transfer by actively selecting the most relevant base classes to novel classes, and then selecting the base domains (*i.e.*, learners) with accurate prediction in a new domain (see Fig. 2(b)). We evaluate AS3 on META-DATASET [41], *i.e.*, a large-scale benchmark that contains diverse datasets and presents more realistic tasks. Extensive experiments have demonstrated the advantages of our proposal both in accuracy and efficiency.

Of note, our main contribution is not to propose negative transfer, but to challenge the basic assumption of multi-domain few-shot classification that more base domains are always better for a few-shot task. In summary, the main contributions of this work include: 1) We present the first systematical investigation on whether it is better in multi-domain few-shot classification to employ as more base classes and domains as possible for adaptation to a new domain; 2) Inspired by the biological memory encoding and retrieval mechanism, we propose a novel data-driven approach to actively silence the redundant base classes and learners that interfere with the learning of a new few-shot task; 3) Thanks to the hierarchical subset selection of base classes and domains, our method achieves over 100% acceleration while maintaining/improving accuracy.

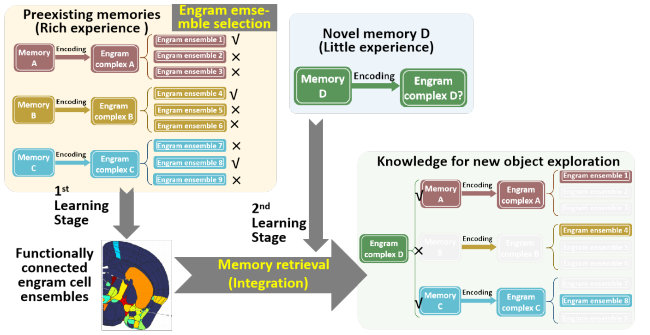
2 RELATED WORK

2.1 Single-Domain Few-Shot Classification

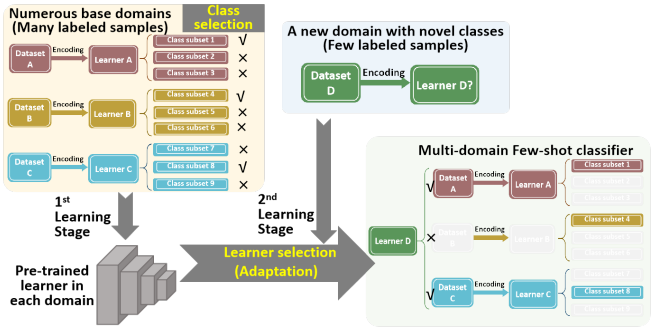
Single-domain few-shot classification aims to recognize samples from several novel classes given only a few labeled samples, relying on **an** extra labeled dataset from other classes (*aka.* base classes). A wide variety of advanced methods have been proposed and significantly improved the few-shot classification accuracy on benchmark datasets (*e.g.*, miniImageNet [5]). In general, these typical methods can be roughly divided into three groups, *i.e.*, fine-tuning based methods [5, 7, 26, 39, 46], meta-learning based methods [12, 15, 23, 30, 32, 36, 37, 45], and metric-learning based methods [17, 20, 24, 26, 34, 35, 38, 42, 44]. A detailed discussion is provided in Appendix as we mainly focus on multi-domain few-shot classification in this work.

2.2 Multi-Domain Few-Shot Classification

By contrast, multi-domain few-shot classification relies on **numerous** extra large labeled datasets from diverse base domains, rather than one. It is recently developed by Triantafillou *et al.* [41] who propose a new benchmark, META-DATASET, and meanwhile, highlight some challenges that current single-domain few-shot learning methods face in the multi-domain setting. Crucially, they find that the methods trained on all available base domains would normally obtain improvements on some new domains at the expense of others. Following on their work, progress has been made, which includes the design of adapted hyper-parameter optimization strategy (*e.g.*, SimpleCNAPS [2]), and more flexible meta-learning based algorithms (*e.g.*, CNAPs [31], MetaNorm [8], and TaskNorm [4]). The most notable is SUR (Selecting Universarial Representation) [9] that extracts a universal representation from a set of domain-specific backbones. In particular, SUR prescribes a feature-selection procedure to weight each backbone and then produce an adapted representation for a new few-shot task. However, except for the underlying universal representation, there is no transfer learning performed with regard to how classification rules are inferred across tasks and domains. To explore this question, based on SUR, Liu *et al.* [27] design a Universal Representation Transformer (URT) layer, which learns to retrieve/blend the appropriate backbones for a new few-shot task. Empirical results show URT sets a new baseline on META-DATASET. To further refine the universal representation for novel classes, Li *et al.* [25] propose to learn a single set of universal



(a) Biological memory encoding and retrieval mechanism, in which brain selects first those relevant memories (*i.e.*, memory complexes) to the retrieval memory, and then the relevant engram ensembles in relevant memories.



(b) Multi-domain few-shot classification via hierarchical subset selection, in which machine selects first those relevant base classes, and then the relevant base datasets (*i.e.*, domains).

Figure 2: Functional consistency between biological memory system and our AS3 to learn new concepts from small samples. Of note, machine can also first select base datasets as brain, when the number of base datasets is really large as that of memories.

deep representations by distilling the knowledge of multiple separately trained networks. For more parameter efficiency, scalability, and adaptability, few-shot learning with a universal template is further developed [40], which fine-tunes its proposed initialization with a few steps of gradient descent.

Of note, most above methods are model-driven, with the focus on designing various training strategies to boost few-shot classification accuracy. By contrast, our work is data-driven for both efficient and accurate multi-domain few-shot classification by actively avoiding redundant pre-training. The most related work to ours is [33, 50] which clearly show that carefully selected base classes can lead to much better accuracy in the *single-domain* setting. However, it is indeed challenging to extend them in the multi-domain setting. Specifically, the base class selection problem in [50] is formulated as a submodular optimization program with cubic complexity with respect to the number of base classes, while our proposal is with linear complexity (see Sec. 4.1). Besides, the query samples are involved in class selection [33], which is not applicable in realistic scenarios. Notably, both of them rely on sample features from a pre-trained network, while we just need class names for class selection. In summary, our proposal enjoys a computational benefit in terms of running time and scales up to a realistic large-scale case with numerous base domains.

3 PRELIMINARY: SPARSE SUBSET SELECTION

As shown in Fig. 3, given two sets $U = \{u_1, u_2, \ldots, u_{|U|}\}$ and $V = \{v_1, v_2, \ldots, v_{|V|}\}$, the sparse subset selection algorithm aims to find a subset $S \subset U$ that can well represent V by minimizing the following cost function

$$\mathcal{L}(S) = \underbrace{\sum_{j=1}^{|V|} \min_{u_i \in S} d_{ij}}_{\text{total cost}} + \lambda \underbrace{|S|}_{\text{subset size}}, \tag{1}$$

where d_{ij} is the distance between u_i and v_j. A smaller d_{ij} indicates a better representation of v_j by u_i. $\min_{u_i \in S} d_{ij}$ is the cost of representing v_j by S. The first term in Eq. (1) is the total representation cost of V by S, and the second term is the size of S. $\lambda \geq 0$ controls the trade-off between the two terms. In general, it is expected to find a small subset with the lowest total cost.

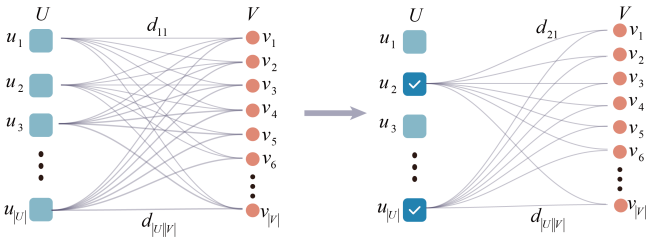


Figure 3: An example of sparse subset selection. Left: All the elements of U are employed to represent the set V. Right: The sparse subset selection algorithm finds a few representative elements of U to well represent the set V. Adapted from [47].

Since $\mathcal{L}(S)$ involves counting the number of elements in subset S, minimizing $\mathcal{L}(S)$ is a discrete optimization problem, and NP-hard. To tackle this issue, we consider an optimization program on unknown variables z_{ij} associated with distance d_{ij}. In particular, $z_{ij} \in \{0, 1\}$ is interpreted as the indicator of u_i representing v_j, which is 1 when u_i is employed to represent v_j, and is 0 otherwise. To ensure that each v_j is represented by exact one element in U, we must have $\sum_{i=1}^{|U|} z_{ij} = 1$. With this notation, we propose to solve

$$\begin{aligned} \min_{\mathbf{Z}} \quad & \overbrace{\sum_{j=1}^{|V|} \sum_{i=1}^{|U|} d_{ij} z_{ij}}^{\text{total cost}} + \lambda \overbrace{\sum_{i=1}^{|U|} \max_j z_{ij}}^{\text{subset size}} \\ \text{s.t.} \quad & \sum_{i=1}^{|U|} z_{ij} = 1, \ \forall j; \ z_{ij} \in \{0, 1\}, \ \forall i, j, \end{aligned} \tag{2}$$

where $\mathbf{Z} = \{z_{ij}\}_{1 \leq i \leq |U|, 1 \leq j \leq |V|}$. $\sum_{i=1}^{|U|} d_{ij} z_{ij}$ is the cost of representing v_j by subset S. In addition, $\max_j z_{ij} = 1$ if u_i is selected to represent some elements in V. Instead of counting the number of elements in S directly as in $\mathcal{L}(S)$, we use the sum of $\max_j z_{ij}$ to denote the size of S.

To solve problem (2) efficiently, we further relax $z_{ij} \in \{0, 1\}$ to a real number $z_{ij} \in [0, 1]$. Then the integer programming-based formulation in problem (2) is converted into a convex optimization problem. As in [10], we adopt the alternating direction method of multipliers (ADMM) [3, 14] to solve (2). Once we obtain the optimal solution $\mathbf{Z}^*$, then the selected subset $S = \{u_q : \exists j \in \{1, \ldots, |V|\}, z^*_{qj} > 0\}$.

4 PROPOSED APPROACH: AS3

Few-shot classification aims to learn a classifier for a new task with a few labeled samples. In other words, the model is learned from a small training set (*aka.* support set) $\mathcal{S} = \{(x_i^*, y_i^*)\}_{i=1}^{n_S}$, and evaluated on a testing set (*aka.* query set) $\mathcal{Q} = \{(x_j, y_j)\}_{j=1}^{n_Q}$. The (x_i, y_i) represents an image-label pair while the pair $(\mathcal{S}, \mathcal{Q})$ represents a few-shot task that contains C novel classes.

We consider the multi-domain few-shot classification problem as in Fig. 1, which has two stages. At the first learning stage, a learning algorithm receives a large base training set $\mathcal{T} = \{\mathcal{D}_k^b\}_{k=1}^K$, which contains B base classes collected from K base domains ($K > 1$ and $B \gg C$ in general). Importantly, the training set $\mathcal{T}$ and the few-shot task $(\mathcal{S}, \mathcal{Q})$ have no categories in common, and are even from totally different distributions. Considering there exist massive redundant, irrelevant and distracted classes in $\mathcal{T}$ for the task $(\mathcal{S}, \mathcal{Q})$, we propose to select a small set of base classes in $\mathcal{T}$ that can well represent novel classes in $(\mathcal{S}, \mathcal{Q})$ (see Sec. 4.1). This can not only reduce the computational cost during the first learning stage, but also silence the non-positive transfer to $(\mathcal{S}, \mathcal{Q})$. Then, only the selected classes in $\mathcal{T}$ are used to learn a set of K feature extractors $\{f_{\theta_k}(\cdot)\}_{k=1}^K$, where $\{\theta_k\}_{k=1}^K$ denotes K base learners.

At the second learning stage, each base learner serves as a feature extractor that maps an input x from $(\mathcal{S}, \mathcal{Q})$ to a d-dimensional representation $f_{\theta_k}(x) \in \mathbb{R}^d$. For the j-th class in $(\mathcal{S}, \mathcal{Q})$, we build a class centroid c_j^k by averaging support samples belonging to this class using learner θ_k:

$$c_j^k = \frac{1}{|\mathcal{S}_j|} \sum_{i \in \mathcal{S}_j} f_{\theta_k}(x_i^*), \quad \mathcal{S}_j = \{q : y_q^* = j\}, \tag{3}$$

where $j = 1, \ldots, C$ and $k = 1, \ldots, K$. We consider a nearest centroid classifier as in [9], where the likelihood function using learner θ_k on an input x from $(\mathcal{S}, \mathcal{Q})$ is

$$p^k(y = l \mid x) = \frac{\exp(-d(f_{\theta_k}(x), c_l^k))}{\sum_{j=1}^C \exp(-d(f_{\theta_k}(x), c_j^k))}. \tag{4}$$

To classify x using θ_k, we choose a distance function $d(\cdot, \cdot)$ to be negative cosine similarity, and assign x to the closest centroid c_j^k. To build a high-quality few-shot classifier, we further propose to find a subset of base learners that capture different types of semantics in $(\mathcal{S}, \mathcal{Q})$, as detailed in Sec. 4.2. This is equivalent to finding a few base domains that can adequately represent a new domain.

Finally, to classify a new query x, we first obtain its global multi-domain representation $f(x)$ that is the concatenation of representations using the selected learners. Then we assign a query x to the closest global centroid c_j, where c_j is the concatenation of c_j^k that use the selected learners.

4.1 Base Class Selection

In the context of base class selection, U in Sec. 3 refers to all B base classes in the base training set $\mathcal{T}$, V is all C novel classes in the few-shot task $(\mathcal{S}, \mathcal{Q})$. We denote $U = \{b_i\}_{i=1}^B$ and $V = \{t_j\}_{j=1}^C$, where b_i and t_j are a base class name and a novel class name, respectively. To calculate the total representation cost in problem (2), we need to define the distance d_{ij} between b_i and t_j. Thanks to the success of BERT to tackle a broad set of NLP tasks [6], an efficient way to measure d_{ij} is based on the pre-trained BERT model $g(\cdot)$[1]. Specifically, $d_{ij} = 1 - \frac{g(b_i)^\mathrm{T} g(t_j)}{\|g(b_i)\| \|g(t_j)\|}$, where $g(\cdot)$ maps an input class name into a vector. Then the base class selection problem is formulated as

$$\begin{aligned} \min_{\mathbf{Z}} \quad & \sum_{j=1}^{C} \sum_{i=1}^{B} d_{ij} z_{ij} \\ \text{s.t.} \quad & \sum_{i=1}^{B} z_{ij} = 1, \forall j;\ z_{ij} \in [0, 1],\ \forall i, j;\ \sum_{i=1}^{B} \max_j z_{ij} \le \tau, \end{aligned} \tag{5}$$

where $\tau \in [0, B]$ is the desired number of selected base classes. In essence, problem (5) is a convex relaxation of problem (2) via Lagrange multiplier, to minimize the total representation cost given a selection 'budget' τ (detailed in Appendix). We can use ADMM to solve this problem with $O(B \log(B) C) \approx O(B)$ computational time, as detailed in Appendix. Then with any model architecture, the k-th feature extractor $f_{\theta_k}(\cdot)$ (*i.e.*, the k-th base learner θ_k) is learned using the selected base classes in the k-th domain.

To improve applicability, we can also select base classes relying on data, where we compute the class similarity in problem (5) using data (pseudo-)centroids instead of class names (Please refer to Appendix for more comparison). However, introducing BERT is equivalent to replacing redundant and distracted calculation in base domains with prior knowledge, benefiting efficient and accurate few-shot learning. Further, our proposal about base class selection is a general approach to boost the efficiency of multi-domain few-shot classification without accuracy loss (see Fig. 6).

4.2 Base Learner Selection

When applying the subset selection algorithm to base learner selection, U refers to the base learner set $\{\theta_k\}_{k=1}^K$, and V is the support set $\mathcal{S} = \{(x_i^*, y_i^*)\}_{i=1}^{n_s}$. Inspired by [47], our goal is to select a set of base learners with the properties of representativeness, confidence and cooperation. For this end, problem (2) is rewritten as

$$\begin{aligned} \min_{\mathbf{Z}} \quad & \sum_{k=1}^{K} \sum_{i=1}^{n_s} z_{ki} d_{ki} + \alpha_1 \sum_{k=1}^{K} \lambda_k \max_i z_{ki} \\ & + \alpha_2 \sum_{1 \le k < l \le K} \mu_{kl} \max_i z_{ki} \cdot \max_i z_{li} \\ \text{s.t.} \quad & z_{ki} \ge 0,\ \forall k, i; \quad \sum_{k=1}^{K} z_{ki} = 1,\ \forall i, \end{aligned} \tag{6}$$

where the first term is the prediction error of using the selected learners on all support samples, the second term is the sparsity term that prefers a few confident learners, and the third term is

[1] We can also use other models, *e.g.*, XLNET [48] or WordNet [29].

the cooperation term to avoid severe contradiction between the selected base learners. α_1 and α_2 control the trade-off among the three terms. We set $\alpha_1 = \alpha_2 = 0.5\alpha_{\max}$, where $\alpha_{\max}$ is the critical value that will result in selecting only one learner *et al.*[10]. Similar to base class selection, we adopt ADMM to solve problem (6) with $O(K\log(K)n_S) \approx O(K)$ complexity, as detailed in Appendix.

In the first term of problem (6), we use the prediction confidence to measure the prediction cost d_{ki} between a base learner θ_k and a support sample x_i^*. We choose prediction confidence because it is widely used to measure the performance of a machine learning model [1]. Specifically, assume the predicted label distribution (*resp.*, the predicted label) of x_i^* using learner θ_k is $\hat{y_i}^k \in \mathbb{R}^C$ (*resp.*, ${l_i}^k \in \{1, \ldots, C\}$). Then the prediction confidence of θ_k on x_i^*, denoted as m_{ki}, is defined as the largest value in $\hat{y_i}^k$ if x_i^* is predicted correctly (*i.e.*, $l_i^k = y_i^*$), and as zero otherwise. Accordingly, we define $d_{ki} = 1 - m_{ki}$ (where $d_{ki} \in [0, 1]$), as we prefer the base learners with both accurate and confident prediction.

In the second term of problem (6), through introducing a penalty, we encourage the selected learners to be more confident of their accurate prediction. Accordingly, we add λ_k for each learner θ_k, which is defined as its negative log-likelihood on support samples, *i.e.*, $\lambda_k = -\sum_{i=1}^{n_S} \log p^k(l_i^k = y_i^* \mid x_i^*)$. A smaller λ_k indicates a more confident learners and hence tends to be selected by our algorithm.

In the third term of problem (6), to avoid severe contradiction between two learners, we penalize the difference of their predictions. KL-divergence is widely used to measure the difference between two distributions, but it is not symmetric and hence not suitable to measure the prediction difference between two learners.. Therefore, we use the symmetric KL-divergence between predictions on the support set, i.e., $\mu_{kl} = \sum_{i=1}^{n_S}(\mathrm{KL}(\hat{y_i}^k \| \hat{y_i}^l) + \mathrm{KL}(\hat{y_i}^l \| \hat{y_i}^k))/(2n_S)$. A larger value indicates a more severe contradiction between two learners θ_k and θ_l.

5 EXPERIMENTS

In this section, we seek to answer two key questions:
Q1: Is it better to employ as more base classes and domains as possible for adaptation to a new specific domain?
Q2: Can AS3 boost computational efficiency without accuracy loss for multi-domain few-shot classification by base classes and domains selection?

5.1 Datasets and Setup

Datasets. We evaluate our method on META-DATASET, a recent large-scale multi-domain few-shot learning benchmark [41]. It consists of 10 datasets with various data distributions across 10 domains, including natural images (ILSVRC-2012, CUB-200-2011, Fungi, VGG Flower, MSCOCO), hand-written characters (Omniglot, Quick Draw), human-created objects (Traffic Signs, Aircraft), and textures images (Describable Textures). Traffic Sign and MSCOCO are reserved for testing only, while all other 8 datasets (*i.e.*, ILSVRC-2012, Omniglot, Aircraft, CUB-200-2011, Describable Textures, Quick Draw, Fungi, and VGG Flower) have their corresponding training, validation and testing sets, with each class assigned to only one of those sets. All these 8 training sets are collected together to serve as the base training set of META-DATASET, which contains 3144 base classes and involves 8 base domains in total. The testing on META-DATASET refers to 10 domains, including the 8 testing sets above, Traffic Sign and MSCOCO. More details of META-DATASET are provided in [41]. To better study out-of-training-domain behavior, we follow [31] and add 3 additional testing datasets (MNIST, CIFAR10, and CIFAR100). Consequently, we have 13 testing domains, including MNIST, CIFAR10, CIFAR100, as well as the testing set of META-DATASET. Few-shot tasks are sampled from each of these testing domains using varying numbers of classes and shots.

Implementation Details. We build AS3 on ResNet-18 backbone, which is consistent with [9, 27] for a fair comparison. As originally suggested [41], all images are resized to 84×84 resolution. We set the number of selected base classes $\tau \in \{500, 1000, 1500\}$. Of note, for efficient evaluation, we perform base class selection once for 600 sampled few-shot tasks in each testing domain. The training details for each dataset are described in Appendix.

Evaluation Metrics. To report test results on META-DATASET, we follow [41] and perform an independent evaluation for each of the 13 datasets, where 600 few-shot tasks are sampled for evaluation on each dataset. For all our experiments, the mean accuracy (in %) over all test tasks with 95% confidence interval is reported.

In addition, we provide a quantitative evaluation for forward knowledge transfer (FKT), which characterizes the transferability that learning a specific set of base classes has on a given few-shot task. Specifically, after selecting τ base classes, we compute the test accuracy over this task, denoted by accuracy(τ), and then obtain

$$\mathrm{FKT}(\tau) = \mathrm{accuracy(all)} - \mathrm{accuracy}(\tau), \tag{7}$$

where accuracy(all) is the accuracy over this few-shot task using all base classes without selection. The smaller the metric FKT(τ), the better the selected base classes. A negative value of FKT(τ) indicates there exists negative pre-trained knowledge in the remaining base classes. Likewise, on each dataset, we will report the mean value (in %) of FKT(τ) over the 600 sampled few-shot tasks.

5.2 A1: More Is Not Always Better

To answer the above first question (*i.e.*, Q1), we survey existing work that train two types of few-shot classifiers, *i.e.*, using the base training sets of ILSVRC-2012 and META-DATASET respectively. Consequently, 8 related methods, including seven single-domain few-shot classification methods (*i.e.*, k-NN, Finetune, MatchingNet, ProtoNet, fo-MAML, RelationNet, and ProtoMAML) [41], and one multi-domain few-shot classification method (*i.e.*, SUR [9]) are chosen. Notably, META-DATASET includes ILSVRC-2012 that has 712 base classes and involves one base domain. We denote the prediction accuracies using the two types of classifiers as accuracy(712) and accuracy(all). Following our proposed metric FKT(·), we further obtain FKT(712) = accuracy(all) − accuracy(712) for each method. Here, FKT(712) measures the pre-trained knowledge transfer from the 7 base training domains in META-DATASET (*i.e.*, Omniglot, Aircraft, CUB-200-2011, Describable Textures, Quick Draw, Fungi, and VGG Flower) to a testing domain. A negative FKT(712) indicates there exists negative pre-trained knowledge in these 7 base training domains. Fig. 4 presents all the results of 8 methods on 10 testing domains of META-DATASET in terms of FKT(712).

As shown in Fig. 4, when using META-DATASET (*i.e.*, 7 extra base training domains besides ILSVRC-2012), there is a dramatic

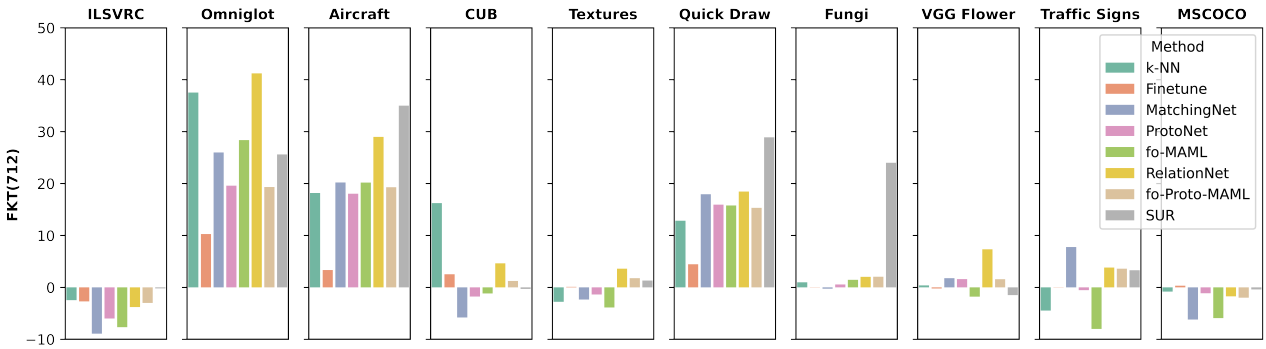


Figure 4: Comparison of 8 methods in terms of forward knowledge transfer (*i.e.*, FKT(712)) that learning an extra 7 base domains has on each testing domain. FKT(712) is de facto accuracy difference of two types of few-shot classifiers using the base training sets of ILSVRC-2012 and META-DATASET, respectively. A negative value/bar indicates there exists negative transfer.

Dataset	k-NN	Finetune	MatchingNet	ProtoNet	fo-MAML	RelationNet	ProtoMAML	CNAPs	TaskNorm	SimpleCNAPs	SUR	URT	AS3	VS.
ILSVRC	38.6±1.0	43.1±1.1	36.1±1.0	44.5±1.1	37.8±1.0	30.9±0.9	46.5±1.1	52.3 ± 1.0	50.6 ± 1.1	58.6 ± 1.1	56.3 ± 1.1	55.7 ± 1.1	56.3 ± 1.1	=
Omniglot	74.6±1.1	71.1±1.4	78.3±1.0	79.6±1.1	83.9±1.0	86.6±0.8	82.7±1.0	88.4 ± 0.7	90.7 ± 0.6	91.7 ± 0.6	93.1 ± 0.5	94.4 ± 0.4	93.9 ± 0.5	=
Aircraft	64.9±0.8	72.0±1.1	69.2±1.0	71.1±0.9	76.4±0.7	69.7±0.8	75.2±0.8	80.5 ± 0.6	83.8 ± 0.6	82.4 ± 0.7	85.4 ± 0.7	85.8 ± 0.6	87.9 ± 0.5	↑
CUB	66.4±0.9	59.8±1.2	56.4±1.0	67.0±1.0	62.4±1.1	54.1±1.0	69.9±1.0	72.2 ± 0.9	74.6 ± 0.8	74.9 ± 0.8	71.4 ± 1.0	76.3 ± 0.8	76.1 ± 1.1	=
Textures	63.6±0.8	69.1±0.9	61.8±0.7	65.2±0.8	64.2±0.8	56.6±0.7	68.3±0.8	58.3 ± 0.7	62.1 ± 0.7	67.8 ± 0.8	71.5 ± 0.8	71.8 ± 0.7	72.1 ±0.8	↑
Quick Draw	44.9±1.1	47.1±1.2	60.8±1.0	64.9±0.9	59.7±1.1	61.8±1.0	66.8±0.9	72.5 ± 0.8	74.8 ± 0.7	77.7 ± 0.7	81.3 ± 0.6	82.5 ± 0.6	80.9 ± 0.6	=
Fungi	37.1±1.1	38.2±1.0	33.7±1.0	40.3±1.1	33.5±1.1	32.6±1.1	42.0±1.2	47.4 ± 1.0	48.7 ± 1.0	46.9 ± 1.0	63.1 ± 1.0	63.5 ± 1.0	64.1 ± 1.6	↑
VGG Flower	83.5±0.6	85.3±0.7	81.9±0.7	86.9±0.7	79.9±0.8	76.1±0.8	88.7±0.7	86.0 ± 0.5	89.6 ± 0.6	90.7 ± 0.5	82.8 ± 0.7	88.2 ± 0.6	84.2 ± 1.0	↓
Traffic Signs	40.1±1.1	66.7±1.2	55.6±1.1	46.5±1.0	42.9±1.3	37.5±0.9	52.4±1.1	56.5 ± 1.1	-	59.2 ± 1.0	53.4 ± 1.0	51.1 ± 1.1	51.1 ± 1.2	↓
MSCOCO	29.6±1.0	35.2±1.1	28.8±1.0	39.9±1.1	29.4±1.1	27.4±0.9	41.7±1.1	42.6 ± 1.1	43.4 ± 1.0	46.2 ± 1.1	52.4 ± 1.1	52.2 ± 1.1	51.7 ± 1.1	=
MNIST	-	-	-	-	-	-	-	92.7 ± 0.4	92.3 ± 0.4	93.9 ± 0.4	94.3 ± 0.4	94.8 ± 0.4	93.3 ± 0.5	=
CIFAR10	-	-	-	-	-	-	-	61.5 ± 0.7	69.3 ± 0.8	74.3 ± 0.7	66.8 ± 0.9	67.3 ± 0.8	67.4 ± 0.9	↓
CIFAR100	-	-	-	-	-	-	-	50.1 ± 1.0	54.6 ± 1.1	60.5 ± 1.0	56.6 ± 1.0	56.9 ± 1.0	56.8 ± 1.0	↓
Average WG	59.2	60.7	64.9	64.9	62.2	58.5	67.5	69.7	71.9	73.8	75.6	77.3	77.0	=
Average SG	-	-	-	-	-	-	-	60.7	-	66.8	64.7	64.5	64.1	=
Average all	-	-	-	-	-	-	-	66.2	-	71.1	71.4	72.3	72.0	=

Table 1: Comparison of AS3 to the previous state-of-the-art approaches on 13 datasets. The few-shot dataset generalization performance is shown in the second group of rows (Traffic Signs - CIFAR100), representing completely out-of-training-domain datasets that require Strong Generalization (SG). For completeness, we also present results on the easier problem of Weak Generalization (WG) in the first 8 rows (ILSVRC - VGG Flower) representing in-of-training-domain datasets. Each number represents the average query set accuracy over 600 test tasks, and its 95% confidence interval. In addition, '-' indicates the accuracy is not reported in original paper. '↑', '=' and '↓' in the column of 'VS.' represent AS3 achieves more promising, comparable and worse accuracy compared with the strongest competitor.

accuracy improvement on Omniglot, Aircraft and Quick Draw testing domains for all compared methods. This is reasonable since the images in these 3 datasets are significantly different from those in ILSVRC-2012. Employing more classes and domains would most likely bring positive knowledge transfer. By contrast, there is no obvious improvement on the remaining 7 testing datasets, even using strong baseline SUR [9]. This might lie in the neglect of data heterogeneity across base and novel classes/domains. As a result, learning 'naively' across all training datasets (*e.g.*, by picking the next dataset to use uniformly at random) does not automatically lead to that desired benefit in most cases. Although exploring heterogeneous data with feature selection, SUR treats all base classes equally that might interfere with the learning of some novel classes. In addition, to do well on K different base domains, base feature extractors must be learned on those base training datasets from scratch. Then stored parameters and computation amount in a few-shot classification method are roughly increased to K times. This just suggests that it is not always better to employ as more base classes and domains as possible for adaptation to a new domain. In particular, we should try to avoid redundant training and non-positive transfer in base training set, for efficient multi-domain few-shot classification.

5.3 A2: Comparison on Accuracy

Next, we answer the above second question (*i.e.*, Q2) in terms of model performance. For this goal, we employ the base training set of META-DATASET that involves 8 base domains, and predict on all 13 testing domains/datasets. In particular, 8 testing datasets among them are in-of-training-domain with weak generalization (WG) from base training set, while other 5 testing datasets are out-of-training-domain with strong generalization (SG) [40]. The competitors include 7 single-domain few-shot classification methods (*i.e.*, k-NN, Finetune, MatchingNet, ProtoNet, fo-MAML, RelationNet, and ProtoMAML) [41], and 5 multi-domain methods (*i.e.*, CNAPs [31], TaskNorm [4], SimpleCNAPs [2], SUR [9] and URT [27]). Here, we fix $\tau = 1500$ in AS3. Tab. 1 provides the comparison of all 12 competitors and our AS3. We observe that even if using half of the base classes, AS3 outperforms the SOTA baselines SUR/URT on 8/4 datasets without compromising much performance on other datasets, while almost consistently outperforms the remaining compared methods. Of note, for out-of-training-domain testing, subset selection is not as advantageous as in in-of-training-domain testing. This may due to the totally different data distribution between base and novel classes.

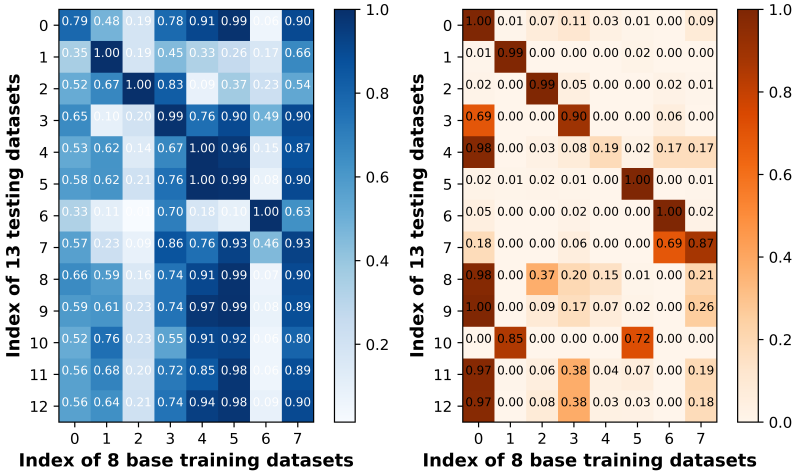


Figure 5: The visualization of selection results by AS3. (Left): Base class selection results about the averaged proportion of selected base classes in each base domain to represent each testing domain; (Right): Base learner selection results about the averaged selection probability of each base domain to represent each testing domain.

To better understand how AS3 selects base classes and learners for adaptation and generalization, in Fig. 5, we visualize the selection results from 8 base domains for 13 testing domains with $\tau = 1500$. Specifically, the element on row t and column k of blue heatmap (see Fig. 5 (Left)) is the averaged proportion of selected base classes in the k-th base training domain, to represent 600 sampled few-shot tasks in the t-th testing domain. It can be concluded that for in-of-training-domain testing cases (*i.e.*, $t \in [0, \ldots, 7]$), AS3 prefers to select base classes from the base domains similar to the testing domain. For instance, when testing on 'Fungi' ($t = 6$), the selected base classes are mainly from 'Fungi', 'CUB' and 'VGG Flower' ($k = \{3, 6, 7\}$). This might be because the pileum of birds in 'CUB' looks like fungi. By contrast, for out-of-training-domain testing cases (*i.e.*, $t \in [8, \ldots, 12]$), AS3 tends to select base classes from all base domains. A similar phenomenon happens in orange heatmap (see Fig. 5 (Right)), where the element on row t and column k is the averaged selection probability of the k-th base domain to represent 600 sampled few-shot tasks in the t-th testing domain. The high sparsity of these probabilities shows that AS3 prefers the base learners with high quality and relevance rather than treating all learners equally. Thereinto, 'Texture' ($t = 4$) is an exception, which may because there exists rich similar textural information in 'ILSVRC' ($k = 0$). The same reason makes most out-of-training-domain testing domains select base domain 'ILSVRC' ($k = 0$). Putting the two heatmaps together, we can find that even if some inappropriate base classes are selected when τ is too large, the selection of base learners can further silence the negative transfer brought by those classes. This is just why AS3 selects domain/learner last.

Further, to explore the effectiveness of base class and learner selection respectively, two own baselines are designed. Specifically, we replace our BERT-based class selection proposal in Sec. 4.1 with random selection, serving as Baseline1, and replace our learner selection in Sec. 4.2 with using all 8 learners equally as Baseline2. In addition, to verify base class selection is general to avoid non-positive transfer, we also reproduce SUR [9] and URT [27] using our selected base classes, denoted as 'SUR w/ sel' and 'URT w/ sel'. For each method, we still report its mean accuracy on 600 few-shot tasks sampled from each testing domain. Fig. 6 shows their accuracy on all 13 testing domains as we change the number of selected classes (*i.e.*, τ). Please refer to Appendix for corresponding comparison in terms of FKT(τ). As the results show, 1) increasing the value of τ, *i.e.*, employing more base classes, might decrease the few-shot classification accuracy except for Baseline1 (*e.g.*, from 84.2% to 82.9% on VGG Flower for $\tau = 1000, 1500$ using AS3). Baseline1 is an exception since random selection is totally irrelevant with the testing domain. Additionally, by selecting only half of base classes, the performance of these methods is higher or quite close to that of using all base classes (*e.g.*, 0.9% improvement on VGG Flower using AS3). This demonstrates again that data is not always the more, the better in multi-domain few-shot classification. We should avoid redundant computation for improving efficiency without accuracy loss. 2) AS3 performs better than two baselines in all the cases, which shows the necessity of both base class and learner selection. 3) Using the selected base classes, SUR [9] and URT [27], in general, can still achieve comparable or even higher accuracy compared to that using all base classes. For instance, URT [27] has an obvious improvement (2%) when using half of base classes. This indicates our base class selection process is indeed general. 4) AS3, in general, achieves comparable or higher accuracy compared to 'SUR w/ sel' and 'URT w/ sel'. This mainly benefits from the active silencing of negative transfer via learner selection.

5.4 A2: Comparison on Efficiency/Practicality

Finally, we continue answer the question 'Q2' in terms of model efficiency. For this end, we provide the computational costs during model training and testing, given a few-shot task with K base training domains. Specifically, a few-shot classification method generally involves three modules during training phase (*i.e.*, pre-processing, backbone building, and adaptation), and inference during testing phase. Compared with SUR and URT, AS3 has an added overhead of base class selection in pre-processing. This includes the complexity of BERT forward ($O(B + C)$), distance computation ($O(BC)$), and solving 5 ($O(B \log(B)C)$), where B and C are the numbers of base and novel classes. Supposing O_b denotes the training complexity of a single backbone, then SUR and URT are with the complexity of KO_b for building backbones, while AS3 is

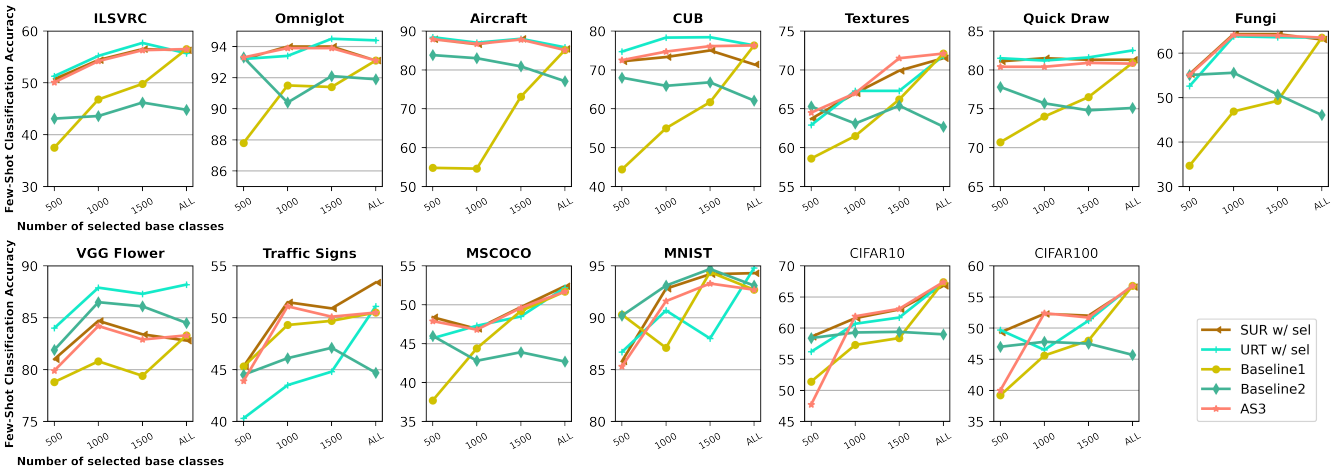


Figure 6: Comparison of AS3 to two adapted SOTA approaches and two own baselines on 13 testing datasets, as a function of the number of selected base classes (*i.e.*, τ). 'ALL' represents using all 3144 base classes in META-DATASET without selection.

with $KO_b/2$ if it selects half of the base classes. For model adaptation, SUR mainly depends on feature selection, being optimized for 40 SGD steps with the complexity of O_f, URT recurs to a transformer layer that is trained for 10,000 episodes with O_t, while our complexity is the lowest with $O(K\log(K))$ using learner selection. Here, $O_f \gg O_t > O(K\log(K))$. Notable, the algorithm complexity for base class and learner selection is really low. Thus, AS3 has the lowest total training cost, although it adds a small overhead in pre-processing. Further, the inference complexity of SUR, URT and AS3 are $O(n_Q)$ since they all rely on distance metrics.

To present the overhead of AS3 more intuitively, we further provide the running time of each part in our algorithm. Concretely, with 12212 MiB GPU memory usage on a single GTX TITAN X, Tab. 2 reports the averaged physical time during model training and inference for a few-shot task. It is concluded that the added overhead on class selection can be neglected in practice to the backbone efficiency and accuracy improvements achieved by selection. In particular, training on a subset of base classes/domains enables the current backbones to converge faster, thus halving backbone training time by halving base classes with 'early stop' as in Tab. 2 . Fig. 7 additionally gives the convergence results on VGG Flower comparing loss values with the time taken. Of note, our main focus is to classify a specific few-shot task (same to baselines (*e.g.*, SUR [9] and URT [27])). With a wider range of unseen classes and domains, *i.e.*, a much larger divergence of base and novel distributions, more base classes/domains may benefit sometimes (as in the out-of-training-domain results in Tab. 1)

		SUR [9]	URT [27]	AS3
Training	Pre-process	-	-	20 *sec.*
	Backbone	2 *days*	2 *days*	1 *day*
	Adaptation	4 *sec.*	11 *min.*	0.38 *sec.*
Testing	Inference	0.1 *sec.*	0.1 *sec.*	0.1 *sec.*

Table 2: Cost comparison on a few-shot task.

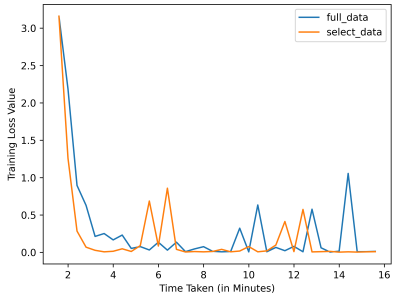


Figure 7: Convergence results on VGG Flower.

6 CONCLUSION

In this work, we investigate redundant non-positive transfer in multi-domain few-shot classification, which largely limits its computational efficiency and prediction accuracy. And then we propose a novel data-driven approach to solve this problem. Thanks to the hierarchical subset selection of base classes and domains, our method achieves over 100% acceleration without accuracy loss. In particular, such a base class selection process is general for improving the efficiency of few-shot classification. Further work will focus on extending our method to other tasks, such as incremental few-shot learning and few-shot object detection.

ACKNOWLEDGMENTS

This work was supported by the National Key Research and Development Program of China (2017YFA0700904, 2020AAA0106000, 2020AAA0104304, 2020AAA0106302, 2021YFB2701000), NSFC Projects (Nos. 62061136001, 62106123, 62076147, U19B2034, U1811461, U19A2081, 61972224), Beijing NSF Project (No. JQ19016), BNRist (BNR2022RC01006), Tsinghua Institute for Guo Qiang, Beijing Academy of Artificial Intelligence (BAAI), Tsinghua-OPPO Joint Research Center for Future Terminal Technology, the High Performance Computing Center, Tsinghua University, and China Postdoctoral Science Foundation (No. 2021T140377, 2021M701892).

REFERENCES

[1] Vineeth Balasubramanian, Shen-Shyang Ho, and Vladimir Vovk. 2014. *Conformal prediction for reliable machine learning: Theory, adaptations and applications*. Newnes.

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

[3] Stephen Boyd, Neal Parikh, and Eric Chu. 2011. *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc.

[4] John Bronskill, Jonathan Gordon, James Requeima, Sebastian Nowozin, and Richard Turner. 2020. Tasknorm: Rethinking batch normalization for meta-learning. In *Proc. ICML*. 1153–1164.

[5] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. 2018. A Closer Look at Few-shot Classification. In *Proc. ICLR*.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. NAACL-HLT (1)*.

[7] Guneet Singh Dhillon, Pratik Chaudhari, Avinash Ravichandran, and Stefano Soatto. 2019. A Baseline for Few-Shot Image Classification. In *Proc. ICLR*.

[8] Yingjun Du, Xiantong Zhen, Ling Shao, and Cees GM Snoek. 2020. Metanorm: Learning to normalize few-shot batches across domains. In *Proc. ICLR*.

[9] Nikita Dvornik, Cordelia Schmid, and Julien Mairal. 2020. Selecting relevant features from a multi-domain representation for few-shot classification. In *Proc. ECCV*. 769–786.

[10] Ehsan Elhamifar, Guillermo Sapiro, and S Shankar Sastry. 2015. Dissimilarity-based sparse subset selection. *IEEE Trans. Pattern Anal. Mach. Intell.* 38, 11 (2015), 2182–2197.

[11] Li Fei-Fei, Rob Fergus, and Pietro Perona. 2006. One-shot learning of object categories. *IEEE Trans. Pattern Anal. Mach. Intell.* 28, 4 (2006), 594–611.

[12] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proc. ICML*. 1126–1135.

[13] Zhiqiang Fu, Yao Zhao, Dongxia Chang, Xingxing Zhang, and Yiming Wang. 2021. Double low-rank representation with projection distance penalty for clustering. In *Proc. CVPR*. 5320–5329.

[14] Daniel Gabay and Bertrand Mercier. 1976. A dual algorithm for the solution of nonlinear variational problems via finite element approximation. *Computers & Mathematics with Applications* 2, 1 (1976), 17–40.

[15] Jonathan Gordon, John Bronskill, Matthias Bauer, Sebastian Nowozin, and Richard E Turner. 2018. VERSA: Versatile and efficient few-shot learning. In *Proc. NeurIPS*. 1–9.

[16] Yunhui Guo, Noel C Codella, Leonid Karlinsky, James V Codella, John R Smith, Kate Saenko, Tajana Rosing, and Rogerio Feris. 2020. A broader study of cross-domain few-shot learning. In *Proc. ECCV*. 124–141.

[17] Jun He, Richang Hong, Xueliang Liu, Mingliang Xu, Zheng-Jun Zha, and Meng Wang. 2020. Memory-augmented relation network for few-shot learning. In *Proc. ACM MM*. 1236–1244.

[18] Yan Hong, Li Niu, Jianfu Zhang, Weijie Zhao, Chen Fu, and Liqing Zhang. 2020. F2gan: Fusing-and-filling gan for few-shot image generation. In *Proc. ACM MM*. 2535–2543.

[19] Sheena A Josselyn and Susumu Tonegawa. 2020. Memory engrams: Recalling the past and imagining the future. *Science* 367, 6473 (2020).

[20] Gregory Koch, Richard Zemel, Ruslan Salakhutdinov, et al. 2015. Siamese neural networks for one-shot image recognition. In *Proc. ICML W*, Vol. 2. Lille, 0.

[21] Brenden Lake, Ruslan Salakhutdinov, Jason Gross, and Joshua Tenenbaum. 2011. One shot learning of simple visual concepts. In *Proc. CogSci*, Vol. 33.

[22] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. 2015. Human-level concept learning through probabilistic program induction. *Science* 350, 6266 (2015), 1332–1338.

[23] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. 2019. Meta-learning with differentiable convex optimization. In *Proc. CVPR*. 10657–10665.

[24] Wenbin Li, Lei Wang, Jinglin Xu, Jing Huo, Yang Gao, and Jiebo Luo. 2019. Revisiting local descriptor based image-to-class measure for few-shot learning. In *Proc. CVPR*. 7260–7268.

[25] Wei-Hong Li, Xialei Liu, and Hakan Bilen. 2021. Universal representation learning from multiple domains for few-shot classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 9526–9535.

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

[27] Lu Liu, William L Hamilton, Guodong Long, Jing Jiang, and Hugo Larochelle. 2020. A Universal Representation Transformer Layer for Few-Shot Image Classification. In *Proc. ICLR*.

[28] Erik G Miller, Nicholas E Matsakis, and Paul A Viola. 2000. Learning from one example through shared densities on transforms. In *Proc. CVPR*. 464–471.

[29] George A Miller. 1995. WordNet: a lexical database for English. *Commun. ACM* 38, 11 (1995), 39–41.

[30] Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. 2020. Rapid Learning or Feature Reuse? Towards Understanding the Effectiveness of MAML. In *Proc. ICLR*.

[31] James Requeima, Jonathan Gordon, John Bronskill, Sebastian Nowozin, and Richard E Turner. 2019. Fast and flexible multi-task classification using conditional neural adaptive processes. In *Proc. NeurIPS*. 7959–7970.

[32] Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. 2019. Meta-Learning with Latent Embedding Optimization. In *Proc. ICLR*.

[33] Othman Sbai, Camille Couprie, and Mathieu Aubry. 2020. Impact of base dataset design on few-shot image classification. In *Proc. ECCV*. 597–613.

[34] Shuai Shao, Lei Xing, Yan Wang, Rui Xu, Chunyan Zhao, Yanjiang Wang, and Baodi Liu. 2021. Mhfc: Multi-head feature collaboration for few-shot learning. In *Proc. ACM MM*. 4193–4201.

[35] Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In *Proc. NeurIPS*. 4080–4090.

[36] Jake Snell and Richard Zemel. 2021. Bayesian Few-Shot Classification with One-vs-Each Pólya-Gamma Augmented Gaussian Processes. In *Proc. ICLR*.

[37] Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele. 2019. Meta-transfer learning for few-shot learning. In *Proc. CVPR*. 403–412.

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

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

[40] Eleni Triantafillou, Hugo Larochelle, Richard Zemel, and Vincent Dumoulin. 2021. Learning a Universal Template for Few-shot Dataset Generalization. *arXiv preprint arXiv:2105.07029* (2021).

[41] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, et al. 2020. Meta-Dataset: A Dataset of Datasets for Learning to Learn from Few Examples. In *Proc. ICLR*.

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

[43] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. 2020. Generalizing from a few examples: A survey on few-shot learning. *Comput. Surveys* 53, 3 (2020), 1–34.

[44] Davis Wertheimer, Luming Tang, and Bharath Hariharan. 2021. Few-Shot Classification With Feature Map Reconstruction Networks. In *Proc. CVPR*. 8012–8021.

[45] Weijian Xu, Huaijin Wang, Zhuowen Tu, et al. 2020. Attentional Constellation Nets for Few-Shot Learning. In *Proc. ICLR*.

[46] Shuo Yang, Lu Liu, and Min Xu. 2021. Free Lunch for Few-shot Learning: Distribution Calibration. In *Proc. ICLR*.

[47] Weikai Yang, Xi Ye, Xingxing Zhang, Lanxi Xiao, Jiazhi Xia, Zhongyuan Wang, Jun Zhu, Hanspeter Pfister, and Shixia Liu. 2022. Diagnosing Ensemble Few-Shot Classifiers. *IEEE Transactions on Visualization and Computer Graphics* 28, 1 (2022), 1–14. https://doi.org/10.1109/TVCG.2022.3182488 To be published..

[48] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Proc. NeurIPS*.

[49] Han-Jia Ye, Xin-Chun Li, and De-Chuan Zhan. 2021. Task Cooperation for Semi-Supervised Few-Shot Learning. In *Proc. AAAI*. 10682–10690.

[50] Linjun Zhou, Peng Cui, Xu Jia, Shiqiang Yang, and Qi Tian. 2020. Learning to select base classes for few-shot classification. In *Proc. CVPR*. 4624–4633.