COMS 4774 Project Report
Unsupervised Few-shot Cross-domain Classification

Han Lin
hl3199@columbia.edu

Xin Lin
xl3120@columbia.edu

Zhexiao Zhang
zz2895@columbia.edu

Abstract

A brief three sentences abstract of our project:
We explored a series of unsupervised and semi-supervised techniques for the purpose of transfer learning with a large domain gap, including teacher-student networks and contrastive learning. We studied their performance empirically with three datasets of different genres. In particular, we investigated various combinations of these techniques and proposed conjectures regarding their efficacy.

1 Introduction

Deep Learning models has achieved empirical successes in many applications. However, their state-of-the-art performance often crucially depends on vast amounts of labeled data, which could be costly to obtain or unavailable for some tasks. Researchers have spent much effort to explore the ways of training models with fewer labels or even no labels at all. A class of methods, called transfer learning [2], tries to store knowledge gained in solving one task and apply it to a different but related task. Another class of methods, called self-supervised learning [8], seeks to eliminate the need of human labeling by leveraging features themselves to produce supervisory signals.

In this project, we are aiming at comparing the performance of different supervised/unsupervised/self-supervised learning methods in the following setting:

- We have access to labeled base class data and unlabeled target class data, as well as only a few shot of labeled target class data.
- The domain gap between base and target class is significant.

To address these two points, we conducted our experiments on the recent BSCD-FSL dataset introduced in year 2020 ([19]), which could be used to test the ability of methods to transfer the knowledge from source dataset (miniImageNet) into three target datasets (CropDisease, EuroSAT, ISIC) which are drastically different from the source dataset (i.e. with large domain gap).

And we attempt to answer the following two questions:

- Is it better to use only the source domain dataset or use only the target domain dataset to train an encoder (representation) for few-shot classification task on target dataset?
- If the encoder (representation) from source domain is still informative, how should we include such information into the representation we need for target domain?

These two question are natural to raise: on the one hand, the unlabeled target dataset is most relavent to the downstream few-shot classification task on the same target dataset, it might be better to use only such unlabeled target dataset to train an representation via self-supervised methods. On the other hand, we might still benefit from the labels in the source domain dataset compared with the unlabeled target dataset, so it might be better to use only the labeled source dataset directly to train a representation. Moreover, somehow combining information from both source and target datasets might be even a better choice. Our project therefore investigated various methods that utilize either the source/target dataset or both to study their performance empirically.

2 Literature Review

2.1 Transfer Learning

To facilitate adaptation across different domains and/or tasks, transfer learning [2] tries to store knowledge gained in solving one task and applying it to a different but related task. It leads to a common paradigm called *pretraining*: a large dataset has been labeled with significant cost (*e.g.* ImageNet [12]), and we try to transfer the knowledge learned on this dataset to various downstream tasks whose labels are scarce or nonexistent (*e.g.* classification on other datasets, or tasks different than whole-image classification such as semantic segmentation).

2.1.1 Few-Shot Learning

In some real-world scenarios such as disease diagnosis with medical images, labeled training samples are often limited. Few-shot learning techniques deal with such challenge of insufficient training data by first training models on a larger base dataset D_{base} and transfer the priors learned to the novel target dataset D_{novel} with fine-tuning. Some correlation between D_{base} and D_{novel} is usually assumed to achieve the best model performance, i.e. the source dataset could be from the same domain as the target dataset. There are 4 typical few-shot learning approaches, each with different assumptions about the knowledge learned from the base dataset. For instance, models could learn a good initialization on the base dataset, so the convergence on the target dataset will be faster provided a few samples from novel classes [15, 16]. Similar to the transfer learning paradigm, the second approach involves learning a discriminative data representation from the in-domain source dataset, assuming that the model parameters trained on the base dataset will also project similar samples in the novel dataset closer in the latent feature space than dissimilar samples [25, 7]. Few-shot meta-learning techniques focus on learning an optimizer on external datasets that efficiently optimize model parameters as it adapt to other specific downstream tasks [36]. Other few-shot learning models aim to transfer knowledge about intra-class variation learned from the base dataset to the novel dataset [35], where the classes in the two datasets are potentially very different in nature. However, these few-shot learning techniques rely greatly on the similarities between the source and target domain. Our project, on the other hand, tries to solve the large domain shift problem.

2.1.2 Cross-domain Few-shot Classification (CD-FSL)

Most existing few-shot learning approaches like metric-based learning have drastically reduced performance when the domain gap is large [9]. Generalizing to a completely new domain with limited training samples remains a big challenge. Adler et al. proposed to combine feature representations at many levels of the neural network when generalizing to the novel dataset [1]. Since the high level abstraction in the last output layer might not be transferable to the target domain given the large domain gap between datasets, adding an ensemble of Hebbian learners could help with the filtering of useful features at different processing level of abstraction, which can be used for any downstream tasks on the novel dataset. Specifically, in computer vision, high level abstractions such as faces might not be useful on target domains like medical imaging, but low level features including edges and curvatures could be transferred to the novel dataset easily. Alternatively, Tseng et al. used a transformation layer before feature encoder to perturb the feature distribution in the source domain and simulate the large domain gap during training [37]. Consequently, the trained feature encoder could adapt to large feature distribution difference in the unseen target dataset at test time.

2.1.3 Representation Learning

Not all features are created equal. Data analysts have long realized the fact that some features are more helpful to prediction than others and regularly engineered better features via various techniques. Representation learning, or feature learning [2], is a family of techniques to automatically discover useful representations that help model performance.

Representation can be learned in a supervised fashion, such as supervised neural networks, supervised dictionary learning. Representations can also be learned in an unsupervised fashion, such as clustering, PCA, matrix factorization, autoencoding, etc. While it is a ubiquitous concept in machine learning, representation learning is explicitly studied in and largely popularized by the recent advances in large-scale deep learning. Often times, representations are learned as a byproduct of the training process, and they are given many names to mark different interpretations assigned to them. For

example, representations are termed feature maps in computer vision, and embeddings or annotations (depending on the representation is the outcome of which layer) in natural language processing.

An Analysis of Single-Layer Networks in Unsupervised Feature Learning: comparison of several feature learning methods ProtoNet learn to produce one prototype for each class (the canonical representation for that class)

Representation learning is closely related to transfer learning because representations, or more precisely the encoders that map inputs to useful representations, are more transportable from one domain to another. It is a common theme in many successful NLP/CV applications: ELMo [31], BERT [13], ResNet [22], ViT [14].

2.2 Knowledge Distillation

Knowledge distillation is originally conceived for model compression. The idea is to train a smaller model (student) to mimic the behavior of a larger model (teacher) by replicating not only the teacher's final prediction but also its intermediate representations of all layers (soft labels). By establishing correspondence between intermediate representations from teacher/student, the student model replicates teacher's behavior but with a smaller model size. Researchers have also argued that soft labels are more accommodative and learning from them facilitate generalization [23] [17]. However, soft labels potentially propagate teacher's errors to students, since they are not as close to golden standard as human annotations.

In our experiment, we adapt the idea of knowledge distillation for transfer purpose: one trains teacher model on the source domain and asks it to predict on on the target domain. Then we use the soft activations instead of hard labels (*i.e.* one-hot encodings) and train a student model to learn from these soft labels. We note that there are also unsupervised way of pseudo labeling. For example, one can use the assignments in clustering as pseudo labels [4].

2.3 Self-Supervised Learning

Self-supervised learning can be thought of as a form of unsupervised learning. Similar to transfer learning, self-supervised learning aims to solve the problem of training models on domains with few data. Instead of transferring knowledge from one domain to another, self-supervised learning works on the target domain directly - one tries to eliminate the need of human labeling by leveraging features to produce supervisory signals.

Self-supervised learning (SSL) can be thought of as a form of unsupervised learning, since it does not make use of human labels, but rather learns from features themselves. Under supervised learning setting, models are trained to approximate the mapping from feature to human labels. In contrast, self-supervised techniques train model to map features to features and obtain useful encoders from it. To avoid trivial solution such as identify mapping, one can apply various forms of regularizations and combinations of them, such as bottlenecks, corruptions, sparsity. There is a wide spectrum of SSL techniques for self-supervised learning, among which autoencoding and contrastive learning are two major families. Depending on the application domain, there also exist other techniques that are more idiosyncratic, such as contrastive predictive coding [30] and cycle consistency correspondence [42].

2.3.1 Autoencoding

Autoencoding is a generative approach usually trained for reconstructing inputs. Autoencoder [24] is originally designed to reconstruct input at output using a neural network with a narrow bottleneck in the middle. and/or some form of regularization to make the recontruction challenging enough: denoising [39], sparse [28], contractive [34].

Masked autoencoding can be thought of as a specific form of denoising, where a portion of the input is excluded from the sight of the model. It is widely applied in NLP as masked token prediction, *e.g.* BERT [13] [26], or in CV as masked patch prediction, *e.g.* ViT [14] [20].

Furthermore, one can also view autoregressive models as a special case of masked autoencoding, where masked portion is decided by temporal ordering rather than random selection. For example, a widely used and empirically successful pretrained objective in NLP is next word prediction: word2vec [27], GPT [33] [3]; and similarly in CV a pretext task called next pixel prediction: PixelCNN [29], PixelRNN [38].

2.3.2 Contrastive Learning

Constrastive learning, as opposed to autoencoding, is a discriminative approach. Since one cannot not use human labels under self-supervised regime, contrastive learning proposes a simple idea of "one versus all", *i.e.* each data point is in its own class against all other data points. To make the contrast challenge enough for encoder to learn something useful, data augmentations are applied to generate different yet highly correlated "views" of the same data points, and one considers these "views" to be in the same class.There are many variants under the umbrella term contrastive learning, often including different data augmentations/transformation and negative sampling techniques, *i.e.* MoCo [21], BYOL [18], SwAV [5], SimCLR [8], Barlow [44], SimSiam [10].

For instance, SimCLR [8] generates two augmented views of a image by cropping/resizing, color distortion and Gaussian blurring. Then they are sent through encoder to form the representation (which will be used for downstream task) and then a small projection head which maps the representation to a low-dimension latent space where a normalized cosine similarity is applied.

Barlow Twins [44] follows a similar line by feeding two augmented versions of samples into the same network for feature extraction. But it uses a different loss learns to make the cross-correlation matrix between these two groups of output features close to the identity. It aims to keep the representation of two views close, while minimizing their redundancy in a similar fashion to PCA whitening.

3 Method

3.1 Few-Shot Cross-Domain Learning Problem formulation

Our unsupervised few-shot cross-domain learning problem is formulated as follows. The domain is defined as a joint distribution P over input space $\mathcal{X}$ and label space $\mathcal{Y}$. A sample pair from this input space and its corresponding label is defined as (x, y). There are two domains in our setting, one is the source domain $(\mathcal{X}_s, \mathcal{Y}_s)$, and the other is the target domain $(\mathcal{X}_t, \mathcal{Y}_t)$, with joint distribution P_s and P_t respectively. Moreover, the domain gap between P_s and P_t are very large in our setting. For example, the source domain we use is from miniImageNet, which contains a diverse image classes of objects, while the target domain we use is from some completely different dataset, like EuroSAT which contains satellite images. The labeled base classes data $\mathcal{D}_{\mathcal{B}} = \{(x_i, y_i)\}_{i=1}^{\mathrm{N}_B} \subset \mathcal{X}_{\mathcal{B}} \times \mathcal{Y}_{\mathcal{B}}$ are sampled from the source domain, while the unlabeled novel classes data $\mathcal{D}_{\mathcal{N}} = \{(x_i)\}_{i=1}^{\mathrm{N}_N} \subset \mathcal{X}_{\mathcal{N}}$ are sampled from the target domain.

Then the learning algorithm will go two stages. In the first stage, the learning algorithm will pre-train its representation $\phi(x)$ which maps input data into $\mathbb{R}^d$ on either D_{base}, or D_{novel}, or a combination of the two. Then in the second phase, the learning algorithms will be able to get access to only a few-shot (1-shot or 5-shot) labeled target domain data, and it need to learn quickly the novel classes and output a classification model f_θ for potential data from target domain. Finally, the performance of this classification model is tested against some new query image from the target domain.

3.2 Evaluation Methods for Few-Shot Cross-Domain Learning

To address the two questions proposed in the introduction section, the following four comparison baselines are tested in our report. Naive Transfer uses only source domain dataset, SimCLR uses only target domain dataset, while Naive Transfer + SimCLR as well as Teacher-Student Network use both data from source and target domain.

3.2.1 Naive Transfer

To answer the question whether is it better to use the source domain representation on target domain directly, we try to apply naive transfer, which simply use the pre-trained encoder from source domain as a feature extractor for target domain, and then train a linear classifier for the target few-shot classification task while keeping the encoder part fixed. The optimization objective function is defined as:

$$\min_{\theta} \frac{1}{\mathrm{N}_B} \sum_{(x_i, y_i) \in \mathcal{D}_B} L_{CE}(f_\theta(x_i), y_i) \tag{1}$$

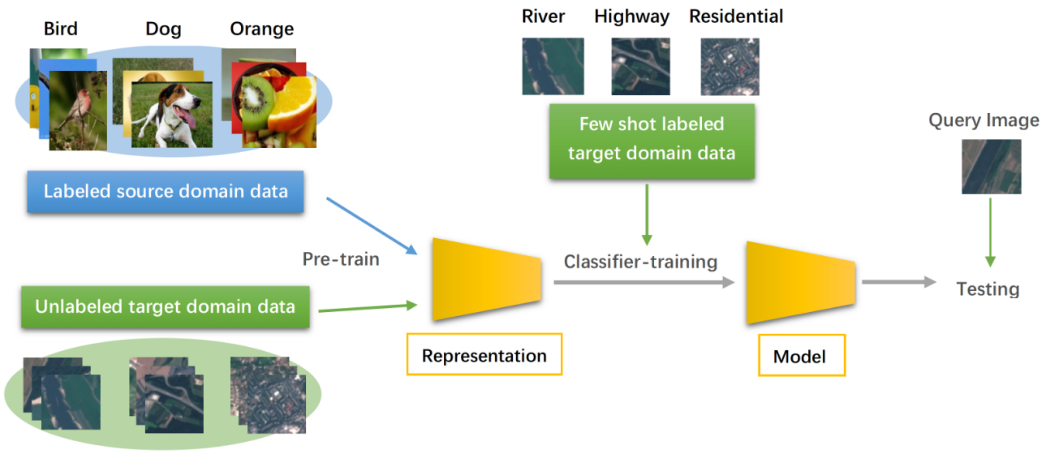

Figure 1: **Few-shot Cross-domain Classification problem formulation.** Our learning algorithms have access to labeled source domain data (miniImageNet) and unlabeled target domain data with large domain difference (CropDisease/EuroSAT/ISIC) to train a representation. Fixed Feature Extractor uses only labeled source domain data; SimCLR uses only unlabeled target domain data; Fixed+SimCLR & Teacher-Student model use both labeled source domain data and unlabeled target domain data. Then the learning algorithms will be able to access only a few-shot (1-shot or 5-shot) labeled target domain data, and need to success in the classification task on queries of new unseen data from the target domain.

where θ is the parameter to be learned, with cross-entropy loss averaged over all samples from base class data from source domain.

3.2.2 Contrastive Learning

To answer the question whether is it better to training a new representation on target domain directly without using base class data, we includes here a contrastive learning approach called SimCLR introduced recently ([8]), which only uses the target domain unlabeled data to train a representation, and then train a linear classifier for the target few-shot classification task.

The SimCLR loss is typical of contrastive learning: it first randomly sample a minibatch of N_N examples and define the contrastive prediction task on pairs of augmented examples derived from the minibatch, resulting in $2\mathrm{N}_N$ data points. Then given a positive pair, it treats the other $2(\mathrm{N}_N - 1)$ augmented examples as negative examples. The loss function for a positive pair of examples (i, j) is defined as:

$$l(i,j) = -\log \frac{\exp(\mathrm{sim}(z_i, z_j)/\tau)}{\sum_{k=1}^{2\mathrm{N}_N} 1_{[k \neq i]} \exp(\mathrm{sim}(z_i, z_k)/\tau)} \tag{2}$$

where $\mathrm{sim}(.)$ is a similarity measure (e.g. cosine similarity) between vector z_i and z_j. Then the optimization objective is to minimize the following loss function by updating model parameters:

$$\min_{\theta} \frac{1}{2\mathrm{N}_N} \sum_{k=1}^{\mathrm{N}_N} [l(2k-1, 2k) + l(2k, 2k-1)] \tag{3}$$

3.2.3 Combining Naive Transfer and Contrastive Learning

The above two methods either uses information only from source domain (naive transfer) or only from target domain (SimCLR) to train the representation. It is natural to ask if it would be beneficial to find a way to combine both domain where we extract useful information from both domains. Therefore, we proposed the following method as another baseline for comparison.

A natural idea to combine information from both source and target domain into a single representation is as follows: we first use train a encoder on the source domain, then we use its weights to initialize the encoder to be trained by SimCLR from the target domain. If the domain gap between source

and target domain is not large, then we would expect the representation not adjusted drastically. Otherwise, if the domain gap is indeed large, then the representation could still be adjusted in a way to put more weight on information from target domain.

3.2.4 Teacher-Student Networks

Inspired by the recent advances from semi-supervised learning (SSL), we also proposed another variant that could also utilize information from both domains. Our method is based on self-training in the teacher-student framework. In the first stage, the teacher model is first trained with labeled data from source domain, and then this trained teacher model is used to generate soft labels for the unlabeled data from target domain. Then in the second stage, the student model is trained on both the labeled source domain data as well as the soft-labeled target domain data with label generated from the teacher. The steps can be formulated as follows:

1. In the first stage, a teacher network with parameter θ_T is trained on the base class from source domain $\mathcal{D}_\mathcal{B}$ by minimizing the cross-entropy loss in Equation (1). Then the teacher network is used to generate pseudo-labels for the unlabeled novel class data $\{(x_i)\}_{i=1}^{\mathrm{N}_N}$ from the target domain $\mathcal{D}_\mathcal{N}$. We denote these pseudo labeled data as $\widehat{y}_i = f_{\theta_T}(x_i), i \in [\mathrm{N}_N]$, and we denote the x,y pairs as the set $\mathcal{D}_\mathcal{N}{}^* = \{(x_i, \widehat{y}_i)\}_{i=1}^{\mathrm{N}_N}$.
2. In the second stage, a new student network with parameter θ_S is learned on both labeled source domain data $\mathcal{D}_\mathcal{B}$ and pseudo labeled target domain data $\mathcal{D}_\mathcal{N}{}^*$ by optimizing the following loss function, where L_{CE} is the cross entropy loss and L_{KL} is the KL divergence.

$$\min_{\theta_S} \frac{1}{\mathrm{N}_B} \sum_{(x_i,y_i)\in\mathcal{D}_B} L_{CE}(f_{\theta_S}(x_i), y_i) + \frac{1}{\mathrm{N}_N} \sum_{(x_i,\widehat{y}_i)\in\mathcal{D}_\mathcal{N}{}^*} L_{KL}(f_{\theta_S}(x_i), \widehat{y}_i) \tag{4}$$

Such method might not be intuitive at first glance, especially why we create soft-labels for students. But if we contemplate on what makes a model robust with good generalization ability to new domains, we would understand the reason for doing this: The soft labels are generated by a softmax function from the last layer of the classifier from the teacher model. Such softmax layer gives each class a probability to be chosen, which contains much more information than the 0/1 hard label usually used in traditional labeling methods. The information contained in each negative class (class that is not the true class) might still be useful, and will therefore increase the robustness of our student model.

To generate such soft-label on unlabeled target domain data, we need to first train a representation (the teacher model) on the labeled source domain data we have at hand, and then use such representation to train a classifier that outputs soft labels for the unlabeled target domain data. Then the student model will be able to use both labeled source and target domain data to generate a new representation that could reflect the information passed from the teacher, as well as information contained in the new target domain.

4 Experiments

4.1 Dataset Description

Our experiments includes datasets from both the base classes of the source domain and the novel classes of the target domain.

Source Domain: The base dataset we used is miniImageNet, which was first introduced in ([41]). It is constructed by choosing 100 random classes from ImageNet, and used the first 80 for training, and the last 20 for testing. Each class includes 600 images of size 84×84. We used this subset of ImageNet dataset because ImageNet is a notoriously large data set which can be quite challenging to run experiments upon it.

Target Domain: The novel dataset in our report includes datasets from challenging benchmark called the Broader Study of Cross-Domain Few-Shot Learning (BSCD-FSL) introduced recently ([19]). This benchmark includes data from CropDiseases, EuroSAT, ISIC, and ChestX datasets, which cover plant disease images, satellite images, dermoscopic images of skin lesions, and X-ray images, respectively. CropDiseases images are natural images specific to agriculture industry. EuroSAT images are less similar as they have lost perspective distortion, but are still color images of natural scenes. ISIC

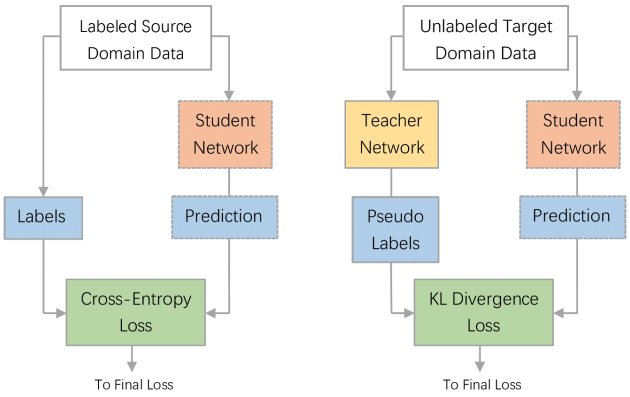

Figure 2: **Diagram of our Teacher Student Framework.** The final loss of our Teacher Student Framework consists of two parts, which are the Cross-Entropy Loss and the KL Divergence Loss. In the left subplot, the Cross-Entropy Loss is calculated between the true labels of the source domain data and the predicted labels from the student network. In the right subplot, the KL Divergence Loss is calculated between the pseudo labels generated from the teacher network and the predicted labels from the student network.

images are even less similar as they have lost perspective distortion and no longer represent natural scenes. ChestX images are the most dissimilar as they have lost perspective distortion, do not represent natural scenes, and is in gray color scale. We tested our models on CropDiseases, EuroSAT as well as ISIC, but not ChestX (45GB), which is too large for us to process.

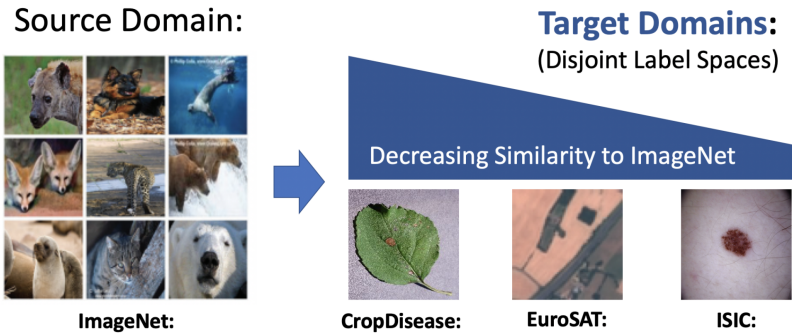

Figure 3: Dataset Description for The Broader Studey of Cross-Domain Few-Shot Learning (BSCD-FSL) benchmark. CropDisease dataset contains perspective natural images, EuroSAT dataset contains natural images with no perspective, and ISIC dataset contains medical images with no perspective. Image source: (Guo et al. [19])

4.2 Implementation Details:

In the representation training stage, we trained with a batch size of 256 on both labeled source domain dataset and unlabeled target dataset for all methods we mentioned in Section 3. We use 5% of the labeled source domain data as well as 10% of the unlabeled target domain data as validation dataset. The learning rate is selected by grid search within $\{0.1, 0.05, 0.03, 0.01, 0.005, 0.003, 0.001\}$ on the validation set. We set the maximum number of training epochs as 1000, with each epoch defined as a complete pass over the training dataset. The training process will stop earlier if the validation loss does not decrease over 100 epochs. We use SGD with 0.9 momentum and 0.0001 weight decay as optimizer. Since the representation training process for Fixed Feature Extractor and the teacher network in Teacher Student Framework is the same, we reuse the Fixed Feature Extractor as the teacher in our framework. All other settings in SimCLR is as default in `https://github.com/sthalles/SimCLR`.

In the finetune stage, we use 5-way 1-shot and 5-way 5-shot to let the model get a glance of labeled target domain data, and then make predictions on 15 unlabeled query images from the target domain. The number of episodes is set as 600.

4.3 Comparison Results:

In this section, we reported the 5-way 1-shot and 5-shot classification accuracy with 95% confidence interval on transfer learning from miniImageNet to CropDisease, EuroSAT and ISIC from ISCD-FSL dataset. We compared methods including Naive Transfer, SimCLR, SimCLR initialized with Naive Transfer, as well as the Teacher-Student semi-supervised method. More comprehensive including additional meta-learning methods are shown in Table 4 in the Appendix.

To make a fair comparison, we use ResNet-10 as encoder backbone across all methods. In the Teacher Student Framework, the first step of training the encoder for teacher's network is the same as Naive Transfer, so we use the encoder from it directly. The result is summarized in Table 1. All methods are trained until the loss are stabilized. Naive Transfer + SimCLR and Teacher Student network tends to be the best among all four methods we compared, which shows that combining information from source and target domain is indeed crutial. Besides, the Teacher Student network performed the best on both EuroSAT and ISIC dataset, and is also the second best for CropDisease dataset under 5-shot learning. Therefore, in the following experiments, we will focus on several variants of such Teacher Student networks to have a better understanding of this method.

Table 1: 5-way 1-shot and 5-shot classification accuracy with 95% CI on transfer learning from miniImageNet to CropDisease, EuroSAT and ISIC from ISCD-FSL dataset. We compared methods including Naive Transfer, SimCLR, SimCLR initialized with Naive Transfer, as well as the Teacher-Student semi-supervised method. The best accuracy is in boldface, and the second best is underscored.

5-way 1-shot			
Dataset	CropDisease	EuroSAT	ISIC
Naive Transfer	$69.82 \pm 0.86\%$	$\underline{60.02 \pm 0.84\%}$	$32.39 \pm 0.61\%$
SimCLR	$\underline{74.76 \pm 0.89\%}$	$42.26 \pm 0.82\%$	$25.87 \pm 0.51\%$
Naive Transfer+SimCLR	$\mathbf{77.03 \pm 0.86\%}$	$58.52 \pm 0.86\%$	$\underline{33.07 \pm 0.61\%}$
Teacher-Student	$74.75 \pm 0.88\%$	$\mathbf{62.23 \pm 0.84\%}$	$\mathbf{33.99 \pm 0.67\%}$
5-way 5-shot			
Dataset	CropDisease	EuroSAT	ISIC
Naive Transfer	$90.16 \pm 0.51\%$	$\underline{79.04 \pm 0.61\%}$	$45.48 \pm 0.63\%$
SimCLR	$91.73 \pm 0.48\%$	$56.26 \pm 0.71\%$	$35.96 \pm 0.56\%$
Naive Transfer+SimCLR	$\mathbf{93.16 \pm 0.43\%}$	$78.44 \pm 0.63\%$	$\underline{46.63 \pm 0.61\%}$
Teacher-Student	$\underline{92.48 \pm 0.48\%}$	$\mathbf{81.94 \pm 0.60\%}$	$\mathbf{47.30 \pm 0.64\%}$

4.4 Ablation of Initialization Schemes of Teacher Student Framework:

In the method Fixed+SimCLR we discussed above, the representation in the second SimCLR stage is initialized with the representation trained from Fixed Feature Extractor, which is a way to combine information from both source and target domain. And our result from table 1 shows that such combination is indeed beneficial for CropDisease and ISIC dataset. However, recent researchers [43] found that initializing the representation of student networks randomly could increase the robustness and generalization ability of the model, which is caused by including noise from such random initialization. Therefore, in this section, we are aiming to delving into this problem and to check whether it is better or not to reuse information of teacher model during training of our student network.

There are two parts of information that could be reused in the student network. The first is the representation from teacher's network, and the second is the classifier parameters from the teacher's network. Since the classes of target domain is different from the classes from source domain, parameters from classifier of teacher's network could only be used to initialize the student's classifier for the source domain data (which is the classifier we used to get predicted label in the left subplot of Figure 2). After the representation of the teacher network is trained, we could use the following four initialization strategies for the representation and classifier of the student network:

(1) **Encoder + Classifier**: Reuse both teacher's encoder and classifier.

(2) **No Encoder + Classifier**: Reuse teacher's classifier but train a new encoder.

(3) **Encoder + No Classifier**: Reuse teacher's encoder but train a new classifier.

(4) **No Encoder + No Classifier**: Train both encoder and classifier from scratch.

Table 2: Ablation test over different initialization schemes for encoder and classifier for student network. 5-way 1-shot and 5-shot classification accuracy with 95% CI on transfer learning from miniImageNet to CropDisease, EuroSAT and ISIC from ISCD-FSL dataset are reported. The best accuracy is in boldface, and the second best is underscored.

5-way 1-shot			
Dataset	CropDisease	EuroSAT	ISIC
Encoder + Classifier	$70.25 \pm 0.88\%$	$60.16 \pm 0.81\%$	$\underline{33.73 \pm 0.65\%}$
No Encoder + Classifier	$69.15 \pm 0.89\%$	$57.82 \pm 0.84\%$	$28.57 \pm 0.56\%$
Encoder + No Classifier	$\underline{72.35 \pm 0.85\%}$	$\mathbf{63.53 \pm 0.85\%}$	$\mathbf{33.99 \pm 0.67\%}$
No Encoder + No Classifier	$\mathbf{74.75 \pm 0.88\%}$	$\underline{62.23 \pm 0.84\%}$	$31.32 \pm 0.60\%$
5-way 5-shot			
Dataset	CropDisease	EuroSAT	ISIC
Encoder + Classifier	$90.58 \pm 0.50\%$	$79.78 \pm 0.64\%$	$\underline{47.10 \pm 0.63\%}$
No Encoder + Classifier	$88.70 \pm 0.56\%$	$76.00 \pm 0.67\%$	$38.76 \pm 0.54\%$
Encoder + No Classifier	$\underline{91.55 \pm 0.48\%}$	$\mathbf{81.94 \pm 0.60\%}$	$\mathbf{47.30 \pm 0.64\%}$
No Encoder + No Classifier	$\mathbf{92.48 \pm 0.48\%}$	$\underline{80.04 \pm 0.67\%}$	$43.20 \pm 0.58\%$

The ablation result is shown in table [5]. As we could see, training a new classifier from scratch for the student network (no classifier) tends to be better for all three datasets, with the only exception that reusing both representation and classifier for ISIC dataset could achieve the second best among all four initialization strategies. Moreover, **encoder + no classifier** achieves the best few-shot classification accuracy for EuroSAT and ISIC dataset, and also the second best for CropDisease, which indicates that reusing the representation tends to be a good choice.

4.5 Ablation test over multiple generations of Student Networks

Our idea to train more generations of Teacher Student networks is motivated by the process of knowledge distillation, which is an effective approach to leverage a well-trained teacher to guide the training of a student network. And after the student network is trained, it is used as a new teacher to train next generation(s) of student network(s). The effectiveness of such process come from the fact that the outputs from the teacher network are used as soft labels for supervising the training of the student network. With more generations, we could expect such soft labels to be softer, which will increase the robustness and generalization ability of our model.

In our experiments shown in Figure 4.5, we reported the vanilla Teacher-Student Network (generation 0) as well as multiple generations from 1 to 5 for the three datasets. 1 shot and 5 shot prediction accuracy were reported. As can be seen, such distillation works well for CropDisease and EuroSAT dataset. Models trained for 3 to 4 generations are the best for CropDisease dataset, and 2 generations are the best for EuroSAT dataset. However, such trend is opposite for ISIC, whose accuracy decreases monotonically with number of generations. Moreover, remember that these three target datasets are sorted in descending order of similarity with the source dataset: CropDisease has lowest domain gap with miniImageNet, while ISIC has the largest domain gap. This is also consistent with their prediction accuracy: CropDisease has highest few-shot accuracy, while ISIC has the lowest.

An intuitive understanding of such result comes from the following trade-off: on the one hand, with more generations, the soft labels could become softer, which will increase the generalization ability of our model. On the other hand, the error will also accumulates with increasing generations of student networks. If the few-shot classification accuracy in the vanilla network (generation 0) is high enough (like CropDisease and EuroSAT), then the advantage from softer labels will dominate. Otherwise, if

the vanilla network cannot classify well on the vanilla network (like ISIC), then the disadvantage from the accumulation of error dominates.

Then we could propose a reasonable conclusion from this experiment: training Teacher Student networks with multiple generations will be a good idea for target dataset with domain shift not too dramatic from the source dataset. And it will behave bad for target dataset that has large domain gap with the source dataset.

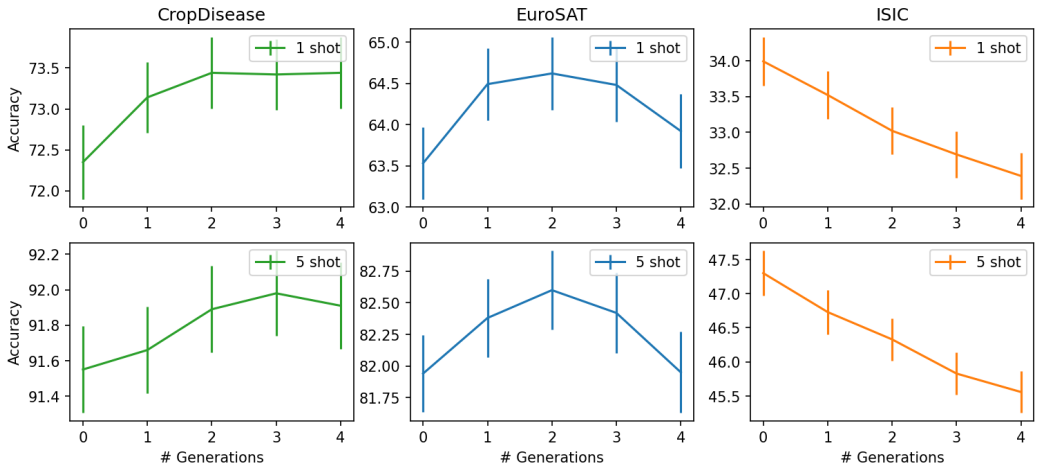

Figure 4: Ablation test over different generations of Student Networks. 0 generation represents the vanilla Teacher Student Network (1 teacher and 1 student). Generations more than 0 means that after the student network is trained, such student is used as a new teacher network again to train the next generation of student networks, etc. 5-way 1-shot and 5-shot classification accuracy with 1 standard deviation on transfer learning from miniImageNet to CropDisease, EuroSAT and ISIC from ISCD-FSL dataset are reported.

4.6 Teacher-Student + Contrastive Loss

An advantage of our Teacher Student Framework is that we could cook up the terms in the final loss function based on our need. In this section, we tried to add another loss term representing contrastive loss (SimCLR and BarlowTwin). The benchmark is the vanilla Teacher Student network with student network initialized by teacher representation and with randomly initialized classifier. And the result is shown in Table 3.

Table 3: Few-shot classification result comparison for Teacher Student with additional contrastive Losses (SimCLR and BarlowTwin). The benchmark is the vanilla Teacher Student network with student network initialized by teacher representation and with randomly initialized classifier. Teacher-Student+SimCLR corresponds to the variants with additional SimCLR Loss in the final loss function, and Teacher-Student+BarlowTwin corresponds to the variants with additional BarlowTwin Loss in the final loss function. Same as before, the best accuracy is in boldface, and the second best is underscored.

5-way 1-shot			
Dataset	CropDisease	EuroSAT	ISIC
Teacher-Student	$72.35 \pm 0.85\%$	$\mathbf{63.53 \pm 0.85\%}$	$33.99 \pm 0.67\%$
Teacher-Student + SimCLR	$74.57 \pm 0.83\%$	$63.03 \pm 0.82\%$	$33.72 \pm 0.64\%$
Teacher-Student + BarlowTwin	$\mathbf{76.34 \pm 0.83\%}$	$62.08 \pm 0.80\%$	$\mathbf{34.44 \pm 0.63\%}$
5-way 5-shot			
Dataset	CropDisease	EuroSAT	ISIC
Teacher-Student	$91.55 \pm 0.48\%$	$81.94 \pm 0.60\%$	$47.30 \pm 0.64\%$
Teacher-Student + SimCLR	$92.44 \pm 0.46\%$	$\mathbf{82.34 \pm 0.60\%}$	$47.36 \pm 0.64\%$
Teacher-Student + BarlowTwin	$\mathbf{93.11 \pm 0.43\%}$	$81.32 \pm 0.62\%$	$\mathbf{47.53 \pm 0.61\%}$

4.7 Ablation test over different number of shots:

Finally in this section, we run different settings of n in 5-way n-shot learning to see the effect of getting more labeled target domain data in the fine-tuning stage. Not surprisingly, the classification accuracy increases as we increase the # shots.

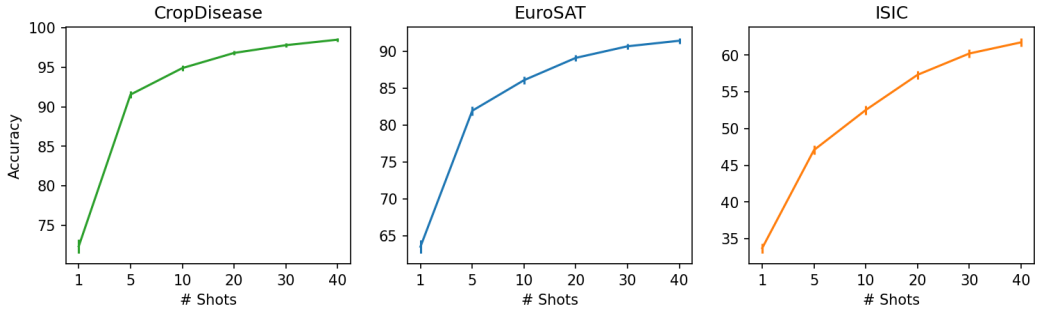

Figure 5: Ablation test over different settings of n in 5-way n-shot learning, with n in $\{1, 5, 10, 20, 30, 40\}$. Classification accuracy with 95% CI on transfer learning from miniImageNet to CropDisease, EuroSAT and ISIC from ISCD-FSL dataset are reported.

5 Discussion

In this final section, we briefly discuss several aspects worth exploring beyond our current project.

5.1 Effect of Representation Backbones

Due to limited computational resource, we use ResNet-10 as backbone to train the representation for all our methods (Fixed Feature Extractor, SimCLR and Teacher Student Framework). Training 300 epochs of Teacher representation already takes around 24 hours. which is the maximum we could do on google colab with a single Tesla-P100 GPU. Generally speaking, backbone with more parameters (ResNet-12, ResNet-18, etc) could increase the classification performance further. Another advantage of our Teacher Student Framework is that it could use other pre-trained representations directly. For example, the teacher network could be imported directly from the publicly available pre-trained representation with ResNet-18 as backbone from ImageNet, which is a much larger and comprehensive dataset.

5.2 Representation with Better Generalization Ability

In Table 5, we tested different initialization schemes for the student network, where the teacher's representation used is the last layer of the neural network, which is very specialized to classify the source domain dataset. There are also other ways to achieve this. For example, the first several layers of the neural network trained for a representation usually contains high lever abstraction not specific to the target domain [1]. Therefore, for further ablation, it might be a good idea to test whether using the first several layers rather than just the last layer could achieve better generalization ability.

5.3 Effect of Self-Supervised Loss

In our experiments, adding SimCLR to Teacher Student networks boosts performance, which empirically proves the efficacy of self-supervisory signals. Researchers have conjectured that the specific form of data augmentation, the loss function and their association with downstream task is critical in the performance of contrastive learning. For example, cropping by itself often does not provide two views that are challenging enough for the model to learn a useful encoder, because models often find the shortcut of identifying the pair by the color distribution. Therefore, a composition of cropping and color distortion proves to be quite effective. The effectiveness of the contrastive task is also closed related to the downstream task where the model is ultimately evaluated on. For instance, distortion

might be a sensible augmentation for classifying satellite images into residential/business/river, but not so much for pose estimation that involves physical laws. Consequently, a interesting direction for future exploration might be customizing the contrastive task, or more generally, the self-supervising objective, by studying the characteristics of the downstream task.

6 Appendix

6.1 Few shot classification results with additional meta-learning methods

In this section, we reported other meta-learning methods in addition to our 4 methods listed in the main body for 5-way 5-shot classification accuracy on the ISCD-FSL datasets. This result is from Table 1 of [19]. We list these additional methods in order to confirm that our comparison benchmarks, which are Naive Transfer and SimCLR, are already the best performers among other meta-learning methods.

Table 4: Additional results of meta-learning methods for 5-way 5-shot classification accuracy with 95% CI on transfer learning from miniImageNet to CropDisease, EuroSAT and ISIC from ISCD-FSL dataset.

5-way 5-shot			
Dataset	CropDisease	EuroSAT	ISIC
MatchingNet	$66.39 \pm 0.78\%$	$64.45 \pm 0.63\%$	$36.74 \pm 0.53\%$
MatchingNet+FWT	$62.74 \pm 0.90\%$	$56.04 \pm 0.65\%$	$30.40 \pm 0.48\%$
MAML	$78.05 \pm 0.68\%$	$71.70 \pm 0.72\%$	$40.13 \pm 0.58\%$
ProtoNet	$79.72 \pm 0.67\%$	$73.29 \pm 0.71\%$	$39.57 \pm 0.57\%$
ProtoNet+FWT	$72.72 \pm 0.70\%$	$67.34 \pm 0.76\%$	$38.87 \pm 0.52\%$
RelationNet	$68.99 \pm 0.75\%$	$61.31 \pm 0.72\%$	$39.41 \pm 0.58\%$
RelationNet+FWT	$64.91 \pm 0.79\%$	$61.16 \pm 0.70\%$	$35.54 \pm 0.55\%$
MetaOpt	$68.41 \pm 0.73\%$	$64.44 \pm 0.73\%$	$36.28 \pm 0.50\%$
Naive Transfer	$90.16 \pm 0.51\%$	$\underline{79.04 \pm 0.61\%}$	$45.48 \pm 0.63\%$
SimCLR	$91.73 \pm 0.48\%$	$56.26 \pm 0.71\%$	$35.96 \pm 0.56\%$
Naive Transfer+SimCLR	$\mathbf{93.16 \pm 0.43\%}$	$78.44 \pm 0.63\%$	$\underline{46.63 \pm 0.61\%}$
Teacher-Student	$\underline{92.48 \pm 0.48\%}$	$\mathbf{81.94 \pm 0.60\%}$	$\mathbf{47.30 \pm 0.64\%}$

6.2 Ablation test over multiple generations of Student Networks

In this section, we reported the results of Figure 4.5 in the following table with both classification accuracy and 95% CI explicitly.

Table 5: Ablation test over different generations of Student Networks. 0 generation means that there is only 1 teacher network and 1 student network. Generations more than 0 means that after the student network is trained, such student is used again as a new teacher network to train the next generation of student networks, etc. 5-way 1-shot and 5-shot classification accuracy with 95% CI on transfer learning from miniImageNet to CropDisease, EuroSAT and ISIC from ISCD-FSL dataset are reported. The best accuracy is in boldface, and the second best is underscored.

5-way 1-shot			
Dataset	CropDisease	EuroSAT	ISIC
0 Generation	$72.35 \pm 0.88\%$	$63.53 \pm 0.85\%$	$\mathbf{33.99 \pm 0.67\%}$
1 Generations	$73.14 \pm 0.85\%$	$64.49 \pm 0.86\%$	$33.52 \pm 0.65\%$
2 Generations	$\mathbf{73.44 \pm 0.86\%}$	$\mathbf{64.62 \pm 0.87\%}$	$33.02 \pm 0.64\%$
3 Generations	$73.42 \pm 0.85\%$	$64.48 \pm 0.88\%$	$32.69 \pm 0.63\%$
4 Generations	$\mathbf{73.44 \pm 0.85\%}$	$63.92 \pm 0.88\%$	$32.39 \pm 0.63\%$
5-way 5-shot			
Dataset	CropDisease	EuroSAT	ISIC
0 Generation	$91.55 \pm 0.48\%$	$81.94 \pm 0.60\%$	$\mathbf{47.30 \pm 0.64\%}$
1 Generations	$91.66 \pm 0.48\%$	$82.38 \pm 0.61\%$	$46.73 \pm 0.63\%$
2 Generations	$91.89 \pm 0.48\%$	$\mathbf{82.60 \pm 0.61\%}$	$46.33 \pm 0.61\%$
3 Generations	$\mathbf{91.98 \pm 0.47\%}$	$82.42 \pm 0.62\%$	$45.83 \pm 0.61\%$
4 Generations	$91.91 \pm 0.48\%$	$81.95 \pm 0.63\%$	$45.56 \pm 0.60\%$

References

[1] Thomas Adler, Johannes Brandstetter, Michael Widrich, Andreas Mayr, David Kreil, Michael Kopp, Günter Klambauer, and Sepp Hochreiter. Cross-domain few-shot learning by representation fusion. *arXiv preprint arXiv:2010.06498*, 2020.

[2] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[3] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 2020.

[4] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149, 2018.

[5] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020.

[6] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020.

[7] Haoxing Chen, Huaxiong Li, Yaohui Li, and Chunlin Chen. Multi-level metric learning for few-shot image recognition. *arXiv preprint arXiv:2103.11383*, 2021.

[8] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *CoRR*, abs/2002.05709, 2020.

[9] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. *arXiv preprint arXiv:1904.04232*, 2019.

[10] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758, 2021.

[11] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223. JMLR Workshop and Conference Proceedings, 2011.

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[15] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR, 2017.

[16] Chelsea Finn, Kelvin Xu, and Sergey Levine. Probabilistic model-agnostic meta-learning. *arXiv preprint arXiv:1806.02817*, 2018.

[17] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129(6):1789–1819, 2021.

[18] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.

[19] Yunhui Guo, Noel C. F. Codella, Leonid Karlinsky, John R. Smith, Tajana Rosing, and Rogério Schmidt Feris. A new benchmark for evaluation of cross-domain few-shot learning. *CoRR*, abs/1912.07200, 2019.

[20] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377*, 2021.

[21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[23] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[24] Geoffrey E Hinton and Ruslan R Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.

[25] Leonid Karlinsky, Joseph Shtok, Sivan Harary, Eli Schwartz, Amit Aides, Rogerio Feris, Raja Giryes, and Alex M Bronstein. Repmet: Representative-based metric learning for classification and few-shot object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5197–5206, 2019.

[26] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019.

[27] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[28] Andrew Ng et al. Sparse autoencoder. *CS294A Lecture notes*, 72(2011):1–19, 2011.

[29] Aaron van den Oord, Nal Kalchbrenner, Oriol Vinyals, Lasse Espeholt, Alex Graves, and Koray Kavukcuoglu. Conditional image generation with pixelcnn decoders. *arXiv preprint arXiv:1606.05328*, 2016.

[30] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[31] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. pages 2227–2237, New Orleans, Louisiana, 2018. Association for Computational Linguistics.

[32] Cheng Perng Phoo and Bharath Hariharan. Self-training for few-shot transfer across extreme task differences. *arXiv preprint arXiv:2010.07734*, 2020.

[33] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[34] Salah Rifai, Pascal Vincent, Xavier Muller, Xavier Glorot, and Yoshua Bengio. Contractive auto-encoders: Explicit invariance during feature extraction. In *Icml*, 2011.

[35] Vivek Roy, Yan Xu, Yu-Xiong Wang, Kris Kitani, Ruslan Salakhutdinov, and Martial Hebert. Few-shot learning with intra-class knowledge transfer. *arXiv preprint arXiv:2008.09892*, 2020.

[36] Qianru Sun, Yaoyao Liu, Tat-Seng Chua, and Bernt Schiele. Meta-transfer learning for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 403–412, 2019.

[37] Hung-Yu Tseng, Hsin-Ying Lee, Jia-Bin Huang, and Ming-Hsuan Yang. Cross-domain few-shot classification via learned feature-wise transformation. *arXiv preprint arXiv:2001.08735*, 2020.

[38] Aaron Van Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. In *International Conference on Machine Learning*, pages 1747–1756. PMLR, 2016.

[39] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103, 2008.

[40] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: Is a correction for chance necessary? In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, page 1073–1080, New York, NY, USA, 2009. Association for Computing Machinery.

[41] Oriol Vinyals, Charles Blundell, Timothy P. Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. *CoRR*, abs/1606.04080, 2016.

[42] Xiaolong Wang, Allan Jabri, and Alexei A Efros. Learning correspondence from the cycle-consistency of time. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2566–2576, 2019.

[43] Qizhe Xie, Eduard H. Hovy, Minh-Thang Luong, and Quoc V. Le. Self-training with noisy student improves imagenet classification. *CoRR*, abs/1911.04252, 2019.

[44] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*, 2021.