Instance Credibility Inference for Few-Shot Learning

Yikai Wang[1,4] Chengming Xu[1] Chen Liu[1] Li Zhang[2] Yanwei Fu[1,3,4]*
[1]School of Data Science, Fudan University
[2]Department of Engineering Science, University of Oxford
[3]MOE Frontiers Center for Brain Science, Fudan University
[4]Shanghai Key Lab of Intelligent Information Processing, Fudan University
{yikaiwang19, cmxu18, chenliu18, yanweifu}@fudan.edu.cn, lz@robots.ox.ac.uk

Abstract

Few-shot learning (FSL) aims to recognize new objects with extremely limited training data for each category. Previous efforts are made by either leveraging meta-learning paradigm or novel principles in data augmentation to alleviate this extremely data-scarce problem. In contrast, this paper presents a simple statistical approach, dubbed Instance Credibility Inference (ICI) to exploit the distribution support of unlabeled instances for few-shot learning. Specifically, we first train a linear classifier with the labeled few-shot examples and use it to infer the pseudo-labels for the unlabeled data. To measure the credibility of each pseudo-labeled instance, we then propose to solve another linear regression hypothesis by increasing the sparsity of the incidental parameters and rank the pseudo-labeled instances with their sparsity degree. We select the most trustworthy pseudo-labeled instances alongside the labeled examples to re-train the linear classifier. This process is iterated until all the unlabeled samples are included in the expanded training set, i.e. the pseudo-label is converged for unlabeled data pool. Extensive experiments under two few-shot settings show that our simple approach can establish new state-of-the-arts on four widely used few-shot learning benchmark datasets including miniImageNet, tieredImageNet, CIFAR-FS, and CUB. Our code is available at: https://github.com/Yikai-Wang/ICI-FSL

1. Introduction

Learning from one or few examples is an important ability for humans. For example, children have no problem forming the concept of “giraffe” by only taking a glance from a picture in a book, or hearing its description as looking like a deer with a long neck [58]. In contrast, the most successful recognition systems [20, 42, 14, 16] still highly rely on an avalanche of labeled training data. This thus increases the burden in rare data collection (*e.g.* accident data in the autonomous driving scenario) and expensive data annotation (*e.g.* disease data for medical diagnose), and more fundamentally limits their scalability to open-ended learning of the long tail categories in the real-world.

Motivated by these observations, there has been a recent resurgence of research interest in few-shot learning [10, 43, 46, 53]. It aims to recognize new objects with extremely limited training data for each category. Basically, a few-shot learning model has the chance to access the source/base dataset with many labeled training instances for model training and then is able to generalize to a disjoint but relevant target/novel dataset with only scarce labeled data. A simplest baseline to transfer learned knowledge to the novel set is fine-tuning [57]. However, it would cause severely overfitting as one or a few instances are insufficient to model the data distributions of the novel classes. Data augmentation and regularization techniques can alleviate overfitting in such a limited-data regime, but they do not solve it. Several recent efforts are made in leveraging learning to learn, or meta-learning paradigm by simulating the few-shot scenario in the training process [24]. However, Chen *et al.* [6] empirically argue that such a learning paradigm often results in inferior performance compared to a simple baseline with a linear classifier coupled with a deep feature extractor.

Given such a limited-data regime (one or few labeled examples per category), one of the fundamental problems for few-shot learning is that one can hardly estimate the data distribution without introducing the inductive bias. To address this problem, two types of strategy resort to model the data distribution of novel category beyond traditional *inductive* few-shot learning: (i) semi-supervised few-shot learning (SSFSL) [28, 37, 45] supposes that we can utilize unlabeled data (about ten times more than labeled data) to help to learn the model; furthermore, (ii) transductive inference [18] for few-shot learning (TFSL) [28, 34] assumes

*Corresponding author.

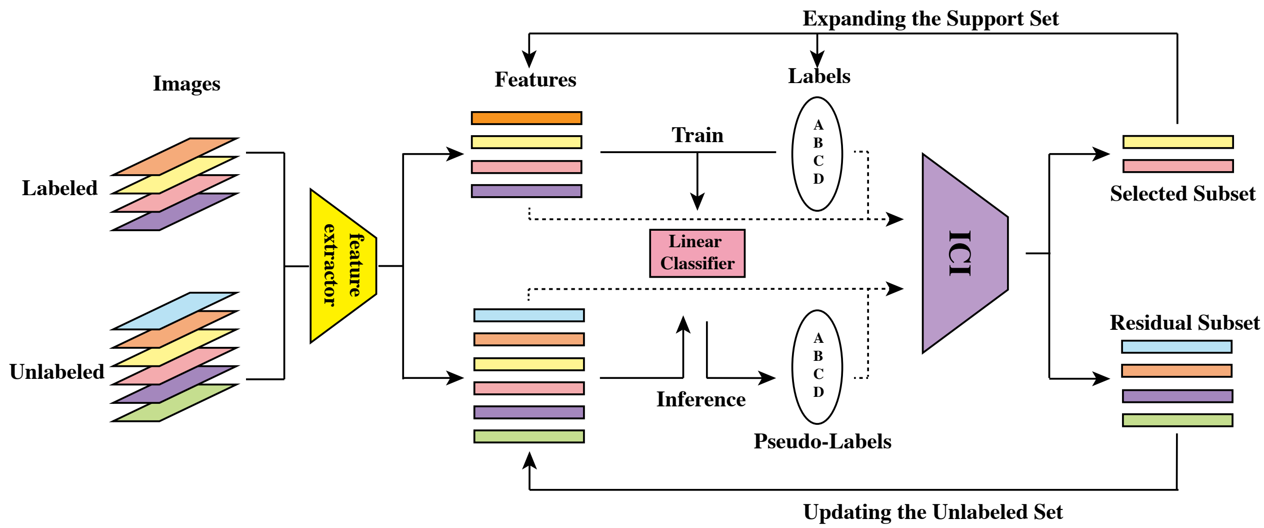

Figure 1. Schematic illustration of our proposed framework. In the inference process of N-way-m-shot FSL task with unlabeled data, we embed each instance, inference each unlabeled data and use ICI to select the most trustworthy subset to expand the support set. This process is repeated until all unlabeled data are included in the support set.

we can access to all the test data, rather than evaluate them one by one in the inference process. In other words, the few-shot learning model can utilize the data distributions of testing examples.

Self-taught learning [35] is one of the most straightforward ways in leveraging the information of unlabeled data. Typically, a trained classifier infers the labels of unlabeled data, which are further taken to update the classifier. Nevertheless, the inferred pseudo-labels may not be always trustworthy; the wrongly labeled instances may jeopardize the performance of the classifier. It is thus essential to investigate the labeling confidence of each unlabeled instance.

To this end, we present a simple statistical approach, dubbed Instance Credibility Inference (ICI) to exploit the distribution support of unlabeled instances for few-shot learning. Specifically, we first train a linear classifier (*e.g*., logistic regression) with the labeled few-shot examples and use it to infer the pseudo-labels for the unlabeled data. Our model aims to iteratively select the most trustworthy pseudo-labeled instances according to their credibility measured by the proposed ICI to augment the training set. The classifier thus can be progressively updated and further infer the unlabeled data. We iterate this process until all the unlabeled samples are included in the expanded training set, *i.e*. the pseudo-label is converged for unlabeled data pool. The schematic illustration is shown in Figure 1.

Basically, we re-purpose the standard self-taught learning algorithm by our ICI algorithm. How to select the pseudo-labeled data to exclude the wrong-predicted samples, *i.e*., excluding the noise introduced by the self-taught learning strategy? Our intuition is that the algorithm of sample selection can neither rely only on the label space (*e.g*. based on the probability of each class given by the classifier) nor the feature space (*e.g*. select samples most similar to training data). Instead, we introduce a linear regression hypothesis by regressing each instance (labeled and pseudo-labeled) from feature to label space and increase the sparsity of the incidental parameter [9] until it vanishes. Thus we can rank pseudo-labeled instances with sparsity degree as their credibility. We conduct extensive experiments on major few-shot learning datasets to validate the effectiveness of our proposed algorithm.

The contributions of this work are as follows: (i) We present a simple statistical approach, dubbed Instance Credibility Inference (ICI) to exploit the distribution support of unlabeled instances for few-shot learning. Specifically, our model iteratively selects the pseudo-labeled instances according to its credibility measured by the proposed ICI for classifier training. (ii) We re-purpose the standard self-taught learning algorithm [35] by our proposed ICI. To measure the credibility of each pseudo-labeled instance, we solve another linear regression hypothesis by increasing the sparsity of the incidental parameter [9] and rank the sparsity degree as the credibility for each pseudo-labeled instance. (iii) Extensive experiments under two few-shot settings show that our simple approach can establish new state-of-the-arts on four widely used few-shot learning benchmark datasets including *mini*ImageNet, *tiered*ImageNet, CIFAR-FS, and CUB.

2. Related work

Semi-supervised learning. Semi-supervised learning (SSL) aims to improve the learning performance with limited labeled data by exploiting large amount of unlabeled data. Conventional approaches focus on finding the low-density separator within both labeled and unlabeled data [52, 4, 18], and avoid to learn the "wrong" knowledge from the unlabeled data [26]. Recently, semi-supervised learning with deep learning models use consistency regularization [21], moving average technique [48] and adversarial perturbation regularization [29] to train the model with large amount of unlabeled data. The key difference between semi-supervised learning and few-shot learning with unlabeled data is that the unlabeled data is still limited in the latter. To some extent, the low-density assumption widely utilized in SSL is hard to achieve in the few-shot scenario, making SSFSL a more difficult problem.

Self-taught learning [35], also known as self-training [55], is a traditional semi-supervised strategy of utilizing unlabeled data to improve the performance of classifiers [1, 12]. Typically, an initially trained classifier predicts class labels of unlabeled instances; the unlabeled data with pseudo-labels are further selected to update the classifier. [22]. Current algorithms based on self-taught learning includes training neural networks using labeled data and pseudo-labeled data jointly [22], using mix-up between unlabeled data and labeled data to reduce the influence of noise [2], using label propagation for pseudo-labeling based on a nearest-neighbor graph and measuring the credibility using entropy [17], and re-weighting the pseudo-labeled data based on the cluster assumption on the feature space [40]. Unfortunately, the predicted pseudo-labels may not be trustworthy. Different and orthogonal to previous re-weighting or mix-up works, we design a statistical algorithm in estimating the credibility of each instance assigned with its corresponding pseudo-label. Only the most confident instances are employed to update the classifier.

Few-shot learning. Recent efforts on FSL are made towards the following aspects. (1) Metric learning methods, putting emphasis on finding better distance metrics, include weighted nearest neighbor classifier (*e.g.* Matching Network [53]), finding prototype for each class (*e.g.* Prototypical Network [43]), or learning specific metric for each task (*e.g.* TADAM [33]); (2) Meta learning methods, such as Meta-Critic [47], MAML [10], Meta-SGD [27], Reptile [32], and LEO [39], optimize the models for the capacity of rapidly adapted to new tasks. (3) Data augmentation algorithms enlarge available data to alleviate the lack of data in the image level [7] or the feature level [37]. Additional, SNAIL [30] utilizes the sequence modeling to create a new framework. The proposed statistical algorithm is orthogonal but potentially useful to improve these algorithms – it is always worth increasing the training set by utilizing the unlabeled data with confidently predicted labels.

Few-shot learning with unlabeled data. Recently approaches tackle few-shot learning problems by resorting to additional unlabeled data. Specifically, in semi-supervised few-shot learning settings, recent works [37, 28] enables unlabeled data from the same categories to better handle the true distribution of each class. Furthermore, transductive settings have also been considered recently. For example, LST [45] utilizes self-taught learning strategy in a meta-learning manner. Different from these methods, this paper presents a conceptually simple statistical approach derived from self-taught learning; our approach, empirically and significantly improves the performance of FSL on several benchmark datasets, by only using very simple classifiers, *e.g.*, logistic regression, or Support Vector Machine (SVM).

3. Methodology

3.1. Problem formulation

We introduce the formulation of few-shot learning here. Assume a base category set $\mathcal{C}_{base}$, and a novel category set $\mathcal{C}_{novel}$ with $\mathcal{C}_{base} \bigcap \mathcal{C}_{novel} = \emptyset$. Accordingly, the base and novel datasets are $D_{base} = \{(\mathbf{I}_i, y_i), y_i \in \mathcal{C}_{base}\}$, and $D_{novel} = \{(\mathbf{I}_i, y_i), y_i \in \mathcal{C}_{novel}\}$, respectively. In few-shot learning, the recognition models on D_{base} should be generalized to the novel category C_{novel} with only one or few training examples per class.

For evaluation, we adopt the standard N*-way-*m*-shot* classification as [53] on D_{novel}. Specifically, in each episode, we randomly sample N classes $L \sim C_{novel}$; and m and q labeled images per class are randomly sampled in L to construct the support set S and the query set Q, respectively. Thus we have $|S| = N \times m$ and $|Q| = N \times q$. The classification accuracy is averaged on query sets Q of many meta-testing episodes. In addition, we have unlabeled data of novel categories $U_{novel} = \{\mathbf{I}_u\}$.

3.2. Self-taught learning from unlabeled data

In general, labeled data for machine learning is often very difficult and expensive to obtain, while the unlabeled data can be utilized for improving the performance of supervised learning. Thus we recap the self-taught learning formalism – one of the most classical semi-supervised methods for few-shot learning [35]. Particularly, assume $f(\cdot)$ is the feature extractor trained on the base dataset D_{base}. One can train a supervised classifier $g(\cdot)$ on the support set S, and pseudo-labeling unlabeled data, $\hat{y}_i = g(f(\mathbf{I}_u))$ with corresponding confidence p_i given by the classifier. The most confident unlabeled instances will be further taken as additional data of corresponding classes in the support set S. Thus we obtain the updated supervised classifier $\hat{g}(\cdot)$. To this end, few-shot classifier acquires additional training instances, and thus its performance can be improved.

However, it is problematic if directly utilizing self-taught learning in one-shot cases. Particularly, the supervised classifier $g(\cdot)$ is only trained by few instances. The unlabeled instances with high confidence may not be correctly categorized, and the classifier will be updated by some wrong instances. Even worse, one can not assume the unlabeled instances follows the same class labels or generative distribution as the labeled data. Noisy instances or outliers may also be utilized to update the classifiers. To this end, we propose a systematical algorithm: Instance Credibility Inference (ICI) to reduce the noise.

3.3. Instance credibility inference (ICI)

To measure the credibility of predicted labels over unlabeled data, we introduce a hypothesis of linear model by regressing each instance from feature to label spaces. Particularly, given n instances of N classes, $S = \{(\mathbf{I}_i, y_i, \mathbf{x}_i), y_i \in \mathcal{C}_{novel}\}$, where y_i is the ground truth when $\mathbf{I}_i$ come from the support set, or the pseudo-label when $\mathbf{I}_i$ come from the unlabeled set, we employ a simple linear regression model to "predict" the class label,

$$y_i = \mathbf{x}_i^\top \beta + \gamma_i + \epsilon_i, \tag{1}$$

where $\beta \in \mathbb{R}^{d\times N}$ is the coefficient matrix for classification; $\mathbf{x}_i \in \mathbb{R}^{d\times 1}$ is the feature vector of instance i; y_i is N dimension one-hot vector denoting the class label of instance i. Note that to facilitate the computations, we employ PCA [50] to reduce the dimension of extracted features $f(\mathbf{I}_i)$ to d. $\epsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$ is the Gaussian noise of zero mean and σ variance. Inspired by incidental parameters [9], we introduce $\gamma_{i,j}$ to amend the chance of instance i belonging to class y_j. Larger $\|\gamma_{i,j}\|$, the higher difficulty in attributing instance i to class y_j.

Write Eq. 1 in a matrix form for all instances, we are thus solving the problem of:

$$\left(\hat{\beta}, \hat{\gamma}\right) = \underset{\beta,\gamma}{\arg\min} \|Y - X\beta - \gamma\|_F^2 + \lambda R(\gamma), \tag{2}$$

where $\|\cdot\|_F^2$ denotes the Frobenius norm. $Y = [y_i] \in \mathbb{R}^{n\times N}$ and $X = \left[\mathbf{x}_i^\top\right] \in \mathbb{R}^{n\times d}$ indicate label and feature input respectively. $\gamma = [\gamma_i] \in \mathbb{R}^{n\times N}$ is the incidental matrix, with the penalty $R(\gamma) = \sum_{i=1}^n \|\gamma_i\|_2$. λ is the coefficient of penalty. To solve Eq. 2, we re-write the function as

$$L(\beta, \gamma) = \|Y - X\beta - \gamma\|_F^2 + \lambda R(\gamma).$$

Let $\frac{\partial L}{\partial \beta} = 0$, we have

$$\hat{\beta} = \left(X^\top X\right)^\dagger X^\top (Y - \gamma), \tag{3}$$

where $(\cdot)^\dagger$ denotes the Moore-Penrose pseudo-inverse. Note that (1) we are interested in utilizing γ to measure the credibility of each instance along its regularization path, rather than estimating $\hat{\beta}$, since the linear regression model is not good enough for classification in general. (2) the $\hat{\beta}$ also relies on the estimation of γ. To this end, we take Eq. 3 into $L(\cdot)$ and solve the problem as,

Algorithm 1 Inference process of our algorithm.

Input:support data$\{(X_i, y_i)\}_{i=1}^{N\times K}$, query data $X_t = \{X_j\}_{j=1}^{M}$, unlabeled data $X_u = \{X_k\}_{k=1}^{U}$
Initialization: support set $(X_s, y_s) = \{(X_i, y_i)\}_{i=1}^{N\times K}$, feature matrix $X_{N\times K+U,d} = [X_s; X_u]$, classifier
Repeat:
Train classifier using (X_s, y_s);
Get pseudo-label y_u for X_u by classifier;
Rank $(X, y) = (X, [y_s; y_u])$ by ICI;
Select a subset (X_{sub}, y_{sub}) into (X_s, y_s);
Until Converged.
Inference:
Train classifier using (X_s, y_s);
Get pseudo-label y_t for X_t by classifier;
Output: inference labels $y_t = \{\hat{y}_j\}_{j=1}^{M}$

$$\underset{\gamma\in\mathbb{R}^{n\times N}}{\arg\min} \|Y - H(Y - \gamma) - \gamma\|_F^2 + \lambda R(\gamma), \tag{4}$$

where $H = X\left(X^\top X\right)^\dagger X^\top$ is the hat matrix of X. We further define $\tilde{X} = (I - H)$ and $\tilde{Y} = \tilde{X}Y$. Then the above equation can be simplified as

$$\underset{\gamma\in\mathbb{R}^{n\times N}}{\arg\min} \left\|\tilde{Y} - \tilde{X}\gamma\right\|_F^2 + \lambda R(\gamma), \tag{5}$$

which is a multi-response regression problem. We seek the best subset by checking the regularization path, which can be easily configured by a blockwise descent algorithm implemented in Glmnet [41]. Specifically, we have a theoretical value of $\lambda_{max} = \max_i \left\|\tilde{X}_{\cdot i}^\top \tilde{Y}\right\|_2 / n$ [41] to guarantee the solution of Eq. 5 all 0. Then we can get a list of λs from 0 to λ_{max}. We solve a specific Eq. 5 with each λ, and get the regularization path of γ along the way. Particularly, we regard γ as a function of λ. When λ changes from 0 to ∞, the sparsity of γ is increased until all of its elements are forced to be vanished. Further, our penalty $R(\gamma)$ encourages γ vanishes row by row, *i.e.*, instance by instance. Moreover, the penalty will tend to vanish the subset of $\tilde{X}$ with the lowest deviations, indicating less discrepancy between the prediction and the ground truth. Hence we could rank the pseudo-labeled data by their λ value when the corresponding γ_i vanishes. As shown in one toy example of Figure 2, the γ value of the instance denoted by the red line vanishes first, and thus it is the most trustworthy sample by our algorithm.

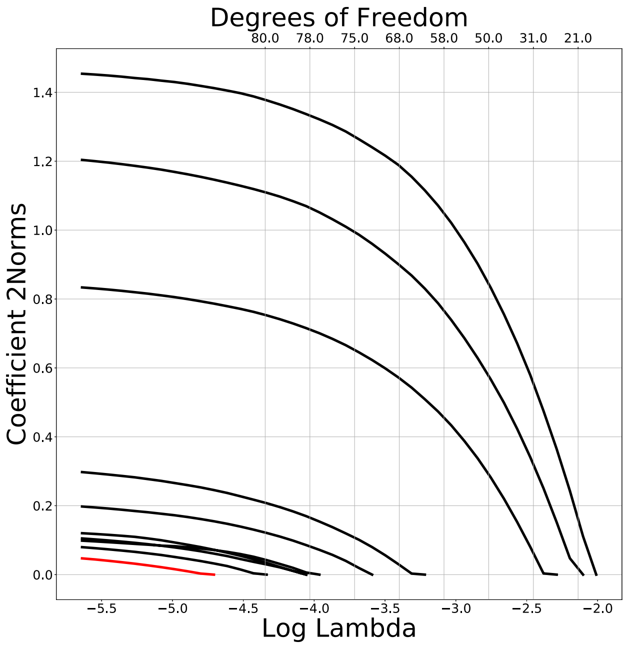

Figure 2. Regularization path of λ on ten samples. Red line is corresponding to the most trustworthy sample suggested by our ICI algorithm.

3.4. Self-taught learning with ICI

The proposed ICI can thus be easily integrated to improve the self-taught learning algorithm. Particularly, the initialized classifier can predict the pseudo-labels of unlabeled instances; and we further employ the ICI algorithm to select the most confident subset of unlabeled instances, to update the classifier. The whole algorithm can be iteratively updated, as summarized in Algorithm 1.

4. Experiments

Datasets. Our experiments are conducted on several widely few-shot learning benchmark datasets for general object recognition and fine-grained classification, including *mini*ImageNet [36], *tiered*ImageNet [37], CIFAR-FS [8] and CUB [54]. ***mini*ImageNet** consists of 100 classes with 600 labeled instances in each category. We follow the split proposed by [36], using 64 classes as the base set to train the feature extractor, 16 classes as the validation set and report performance on the novel set which consists of 20 classes. ***tiered*ImageNet** is a larger dataset compared with *mini*ImageNet, and its categories are selected with hierarchical structure to split base and novel datasets semantically. We follow the split introduced in [37] with base set of 20 superclasses (351 classes), validation set of 6 superclasses (97 classes) and novel set of 8 superclasses (160 classes). Each class contains 1281 images on average. **CUB** is a fine-grained dataset of 200 bird categories with 11788 images in total. Following the previous setting in [15], we use 100 classes as the base set, 50 for validation and 50 as the novel set. To make a fair comparison, we crop all images with the bounding box provided by [51]. **CIFAR-FS** is a dataset with lower-resolution images derived from CIFAR-100 [19] . It contains 100 classes with 600 instances in each class. We follow the split given by [8], using 64 classes to construct the base set, 16 for validation and 20 as the novel set.

Experimental setup. Unless otherwise specified, we use the following settings and implementation in the experiments for our approach to make a fair comparison. As in [30, 33, 23], we use ResNet-12 [13] with 4 residual blocks as the feature extractor in our experiments. Each block consists of three 3×3 convolutional layers, each of which followed by a BatchNorm layer and a LeakyReLu(0.1) activation. In the end of each block, a 2×2 max-pooling layer is utilized to reduce the output size. The number of filters in each block is 64, 128, 256 and 512 respectively. Specifically, referring to [23], we adopt the Dropout [44] in the first two block to vanish 10% of the output, and adopt DropBlock [11] in the latter two blocks to vanish 10% of output in channel level. Finally, an average-pooling layer is employed to produce the input feature embedding. We select 90% images from each training class (*e.g.*, 64 categories for *mini*ImageNet) to construct our training set for training the feature extractor and use the remaining 10% as the validation set to select the best model. We use SGD with momentum as the optimizer to train the feature extractor from scratch. Momentum factor and L_2 weight decay is set to 0.9 and $1e-4$, respectively. All inputs are resized to 84×84. We set the initial learning rate of 0.1, decayed by 10 after every 30 epochs. The total training epochs is 120 epochs. In all of our experiments, we normalize the feature with L_2 norm and reduce the feature dimension to $d = 5$ using PCA [50]. Our model and all baselines are evaluated over 600 episodes with 15 test samples from each class.

4.1. Semi-supervised few-shot learning

Settings. In the inference process, the unlabeled data from the corresponding category pool is utilized to help FSL. In our experiments, we report following settings of SSFSL: (1) we use 15 unlabeled samples for each class, the same as TFSL, to compare our algorithm in SSFSL and TFSL settings; (2) we use 30 unlabeled samples in 1-shot task, and 50 unlabeled samples in 5-shot task, the same as current SSFSL approaches [45]; (3) we use 80 unlabeled samples, to show the effectiveness of ICI compared with FSL algorithms with a larger network and higher-resolution inputs. We denote these as (15/15), (30/50) and (80/80) in Table 1. Note that CUB is a fine-grained dataset and does not have

Setting	Model	*mini*ImageNet 1shot	*mini*ImageNet 5shot	*tiered*ImageNet 1shot	*tiered*ImageNet 5shot	CIFAR-FS 1shot	CIFAR-FS 5shot	CUB 1shot	CUB 5shot
In.	Baseline* [6]	51.75	74.27	-	-	-	-	65.51	82.85
	Baseline++* [6]	51.87	75.68	-	-	-	-	67.02	83.58
	MatchingNet* [53]	52.91^1	68.88^1	-	-	-	-	72.36^1	83.64^1
	ProtoNet* [43]	54.16^1	73.68^1	-	-	72.20^3	83.50^3	71.88^1	87.42^1
	MAML* [10]	49.61^1	65.72^1	-	-	-	-	69.96^1	82.70^1
	RelationNet* [46]	52.48^1	69.83^1	-	-	-	-	67.59^1	82.75^1
	adaResNet [31]	56.88	71.94	-	-	-	-	-	-
	TapNet [56]	61.65	76.36	63.08	80.26	-	-	-	-
	CTM† [25]	64.12	80.51	68.41	84.28	-	-	-	-
	MetaOptNet [23]	64.09	80.00	65.81	81.75	72.60	84.30	-	-
Tran.	TPN [28]	59.46	75.65	58.68^4	74.26^4	65.89^4	79.38^4	-	-
	TEAM* [34]	60.07	75.90	-	-	70.43	81.25	80.16	87.17
Semi.	MSkM with MTL	62.10^2	73.60^2	68.6^2	81.00^2	-	-	-	-
	TPN with MTL	62.70^2	74.20^2	72.10^2	83.30^2	-	-	-	-
	MSkM [37]	50.40	64.40	52.40	69.90	-	-	-	-
	TPN [28]	52.78	66.42	55.70	71.00	-	-	-	-
	LST [45]	70.10	78.70	77.70	85.20	-	-	-	-
In.	LR	56.06	75.70	69.02	85.37	62.25	80.82	76.16	90.32
In.	SVM	54.46	74.76	67.51	84.67	60.94	79.93	75.84	89.26
Tran.	LR + ICI	66.80	79.26	80.79	87.92	73.97	84.13	88.06	92.53
Tran.	SVM + ICI	65.77	78.94	80.56	87.93	73.16	83.72	87.87	92.38
Semi.	SVM + ICI (15/15)	64.81	78.11	79.72	87.39	72.52	83.23	86.83	91.58
Semi.	SVM + ICI (30/50)	68.24	79.25	83.14	88.58	75.50	84.00	88.94	92.14
Semi.	LR + ICI (15/15)	65.86	78.87	81.10	87.83	73.67	83.85	87.28	92.18
Semi.	LR + ICI (30/50)	69.66	80.11	84.01	89.00	76.51	84.32	89.58	92.48
Semi.	LR + ICI (80/80)	**71.41**	**81.12**	**85.44**	**89.12**	**78.07**	**84.76**	**91.11**	**92.98**

Table 1. Test accuracies over 600 episodes on several datasets. Results with $(\cdot)^1$ are reported in [6], with $(\cdot)^2$ are reported in [45], with $(\cdot)^3$ are reported in [23]. $(\cdot)^4$ is our implementation with the official code of [28]. Methods denoted by $(\cdot)^*$ denotes ResNet-18 with input size 224×224, while $(\cdot)^\dagger$ denotes ResNet-18 with input size 84×84. Our method and other alternatives use ResNet-12 with input size 84×84. **In.** and **Tran.** indicate inductive and transductive setting, respectively. **Semi.** denotes semi-supervised setting where $(\cdot/\cdot)$ shows the number of unlabeled data available in 1-shot and 5-shot experiments.

so sufficient samples in each class, so we simply choose 5 as support set, 15 as query set and other samples as unlabeled set (about 39 samples on average) in the 5-shot task in the latter two settings. For all settings, we select 5 samples for every class in each iteration. The process is finished when at most five instances for each class are excluded from the expanded support set. *i.e.*, select (10/10), (25/45), (75/75) unlabeled instances in total. Further, we utilize Logistic Regression (denoted as *LR*) and linear Support Vector Machine (denoted as *SVM*) to show the robustness of ICI against different linear classifiers.

Competitors. We compare our algorithm with current approaches in SSFSL. TPN [28] uses labeled support set and unlabeled set to propagate label to one query sample each time. LST [45] also uses self-taught learning strategy to pseudo-label data and select confident ones, but they do this by a neural network trained in the meta-learning manner for many iterations. Other approaches include Masked Soft k-Means [37] and a combination of MTL with TPN and Masked Soft k-Means reported by LST.

Results. are shown in Table 1 where denoted as Semi. in the first column. Analysis from the experimental results, we can find that: (1) Compare SSFSL with TFSL with the same number of unlabeled data, we can see that our SSFSL results are only reduced by a little or even beat TFSL results, which indicates that the information we got from the unlabeled data are robust and we can indeed handle the true distribution with unlabeled data practically. (2) The more unlabeled data we get, the better performance we have. Thus we can learn more knowledge with more unlabeled data almost consistently using a linear classifier (*e.g.* logistic regression). When lots of unlabeled data are accessible, ICI achieves state-of-the-art in all experiments even compared with competitors which use bigger network and higher-resolution inputs. (3) Compared with other SSFSL approaches, ICI also achieves varying degrees of improvements in almost all tasks and datasets. These results further indicate the robustness of our algorithm. Compared logistic regression with SVM, the robustness of ICI still holds.

4.2. Transductive few-shot learning

Settings. In transductive few-shot learning setting, we have chance to access the query data in the inference stage. Thus the unlabeled set and the query dataset are the same. In our experiments, we select 5 instances for each class in each iteration and repeat our algorithm until all the expected query samples are included, *i.e.*, each class will be expanded by at most 15 images. We also utilize both Logistic Regression and SVM as our classifier, respectively.

Competitors. We compare ICI with current TFSL approaches. TPN [28] constructs a graph and uses label propagation to transfer label from support samples to query samples and learn their framework in a meta-learning way. TEAM [34] utilizes class prototypes with a data-dependent metric to inference labels of query samples.

Results. are shown in Table 1 where denoted as Tran. in the first column. Experiments cross four benchmark datasets indicate that: (1) Compared with basic linear classifier, ICI enjoys consistently improvements, especially in the 1-shot setting where the labeled data is extremely limited and such improvements are robust regardless of utilizing which linear classifiers. Further, compared results between *mini*ImageNet and *tiered*ImageNet, we can find that the improvement margin is in the similar scale, indicating that the improvement of ICI does not rely on the semantic relationship between base set and novel set. Hence the effectiveness and robustness of ICI is confirmed practically. (2) Compared with current TFSL approaches, ICI also achieves the state-of-the-art results.

4.3. Ablation study

Effectiveness of ICI. To show the effectiveness of ICI, we visualize the regularization path of γ in one episode of inference process in Figure 3 where red lines are instances that are correct-predicted while black lines are wrong-predicted ones. It is obvious that that most of the correct-predicted instances lie in the lower-left part. Since ICI will select samples whose norm will vanish in a lower λ. We could get more correct-predicted instances than wrong-predicted instances in a high ratio.

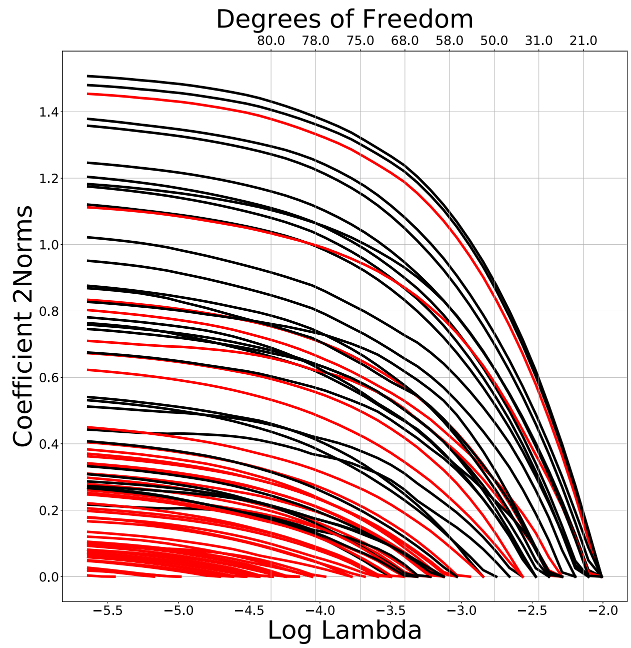

Figure 3. Regularization path of λ. Red lines are correct-predicted instances while black lines are wrong-predicted ones. ICI will choose instances in the lower-left subset.

Model	Tran.		Semi.	
	1shot	5shot	1shot	5shot
LR	56.06	75.43	56.06	75.43
+ ra	59.01	76.38	59.46	76.58
+ nn	63.24	77.63	63.10	77.75
+ co	63.29	77.92	63.57	77.71
ICI	**65.32**	**78.30**	**64.60**	**77.96**

Table 2. Compare to baselines on *mini*ImageNet under several settings.

Compare to baselines. To further show the effectiveness of ICI, we compare ICI with other sample selection strategies under the self-taught learning pipeline. One simple strategy is randomly sampling the unlabeled data into the expanded support set in each iteration, denoted as *ra*. Another is selecting the data based on the confidence given by the classifier, denoted by *co*. In this strategy, the more confident the classifier is to one sample, the more trustworthy that sample is. The last one is replacing our algorithm of computing credibility by choosing the nearest-neighbor of each class in the feature space, denoted as *nn*. In this part, we have 15 unlabeled instances for each class and select 5 to re-train the classifier by different methods for Semi. and Tran. task on *mini*ImageNet. From Table 2, we observe that ICI outperforms all the baselines in all settings.

Effectiveness of iterative manner. Our intuition is the proposed ICI learns to generate a set of trustworthy unla-

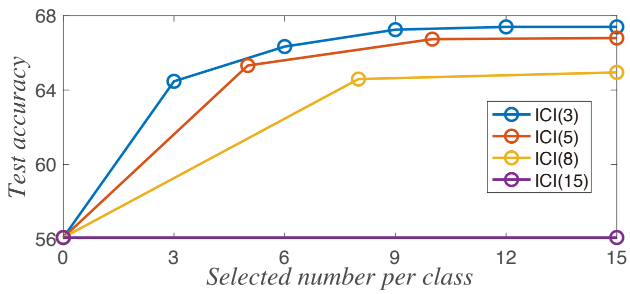

Figure 4. Variation of accuracy as the selected samples increases over 600 episodes on *mini*ImageNet. "ICI (n)": select n samples per class in each iteration.

belled data for classifier training. Select all the unlabelled data in *one go* cannot take the distribution, or the credibility of the unlabeled data into account, and thus produce more noise labels to hurt the performance of the model. The classifier thus be trained with its prediction, resulting in no improvements in TFSL setting. We briefly validate this as ICI (15) in Figure 4 whilst ICI obtained better accuracy with iterative selection manner. For example, select 6 images with two iterations (ICI(3)) is superior to select 8 images in one iteration (ICI(8)).

Acc (%)	0-10	10-20	20-30	30-40	40-50
b/t	0/0	0/0	1/3	16/23	105/125
Acc (%)	50-60	60-70	70-80	80-90	90-100
b/t	193/218	171/189	34/40	2/2	0/0

Table 3. We run 600 episodes, with each episode training an initial classifier. We denote "Acc" as the accuracy intervals; and "b/T" as the number of classifiers experienced improvement v.s. total classifiers in this accuracy interval.

Robustness against initial classifier. What are the requirements for the initial linear classifier? Is it necessary to satisfy that the accuracy of the initial linear classifier is higher than 50% or even higher? The answer is no. As long as the initial linear classifier can be trained, theoretically our method should work. It thus is a future open question of how the initial classifier affects. We briefly validate it in Table 3. We run 600 episodes, with each episode training an initial classifier with different classification accuracy. Table 3 shows that most classifiers can get improved by ICI regardless of the initial accuracy (even with accuracy of 30-40%).

Influence of reduced dimension. In this part, we study the influence of reduced dimension d in our algorithm on 5-way 1-shot *mini*ImageNet experiments. The results with reduced dimension 2, 5, 10, 20, 50, and without dimensionality reduction *i.e.*, $d = 512$, are shown in Table 4. Our algorithm achieves better performance when the reduced

d	Acc (%)	Alg.	Acc (%)
2	63.71 ± 1.025	Isomap [49]	66.53 ± 1.073
5	$\mathbf{66.80 \pm 1.096}$	PCA [50]	66.80 ± 1.096
10	66.25 ± 1.048	LTSA [59]	64.61 ± 1.058
20	64.98 ± 1.049	MDS [5]	59.99 ± 0.941
50	61.54 ± 0.980	LLE [38]	67.59 ± 1.120
512	57.41 ± 0.877	SE [3]	67.70 ± 1.117

Table 4. Influence of dimensionality reduction dimensions and algorithms.

dimension is much smaller than the number of instances (*i.e.*, $d \ll n$), which is consistent with the theoretical property [9]. Moreover, we can observe that our model achieves the best accuracy 66.80% when $d = 5$. Practically, we adopt $d = 5$ in our model.

Influence of dimension reduction algorithms. Furthermore, we study the robustness of ICI to different dimension reduction algorithms. We compare Isomap [49], principal components analysis [50] (PCA), local tangent space alignment [59] (LTSA), multi-dimensional scaling [5] (MDS), locally linear embedding [38] (LLE) and spectral embedding [3] (SE) on 5-way 1-shot *mini*ImageNet experiments. From Table 4 we can observe that ICI is robust across most of the dimensionality reduction algorithms (from LTAS 64.61% to SE 67.7%) except MDS (59.99%). We adopt PCA for dimension reduction in our method.

5. Conclusion

In this paper, we have proposed a simple method, called Instance Credibility Inference (ICI) to exploit the distribution support of unlabeled instances for few-shot learning. The proposed ICI effectively select the most trustworthy pseudo-labeled instances according to their credibility to augment the training set. In order to measure the credibility of each pseudo-labeled instance, we propose to solve a linear regression hypothesis by increasing the sparsity of the incidental parameters [9] and rank the pseudo-labeled instance with their sparsity degree. Extensive experiments show that our simple approach can establish new state-of-the-arts on four widely used few-shot learning benchmark datasets including *mini*ImageNet, *tiered*ImageNet, CIFAR-FS, and CUB.

Acknowledgement. This work was supported in part by NSFC Projects (U1611461,61702108), Science and Technology Commission of Shanghai Municipality Projects (19511120700, 19ZR1471800), Shanghai Municipal Science and Technology Major Project (2018SHZDZX01), and Shanghai Research and Innovation Functional Program (17DZ2260900).

References

[1] Massih-Reza Amini and Patrick Gallinari. Semi-supervised logistic regression. In *ECAI*, 2002.

[2] Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. *arXiv preprint arXiv:1908.02983*, 2019.

[3] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 2003.

[4] Kristin P Bennett and Ayhan Demiriz. Semi-supervised support vector machines. In *NeurIPS*, 1999.

[5] Ingwer Borg and Patrick Groenen. Modern multidimensional scaling: Theory and applications. *Journal of Educational Measurement*, 2003.

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

[7] Zitian Chen, Yanwei Fu, Yu-Xiong Wang, Lin Ma, Wei Liu, and Martial Hebert. Image deformation meta-networks for one-shot learning. In *CVPR*, 2019.

[8] Arnout Devos, Sylvain Chatel, and Matthias Grossglauser. Reproducing meta-learning with differentiable closed-form solvers. In *ICLR*, 2019.

[9] Jianqing Fan, Runlong Tang, and Xiaofeng Shi. Partial consistency with sparse incidental parameters. *arXiv preprint arXiv:1210.6950*, 2012.

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

[11] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. Dropblock: A regularization method for convolutional networks. In *NeurIPS*, 2018.

[12] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *NeurIPS*, 2005.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, 2015.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[15] Nathan Hilliard, Lawrence Phillips, Scott Howland, Artëm Yankov, Courtney D Corley, and Nathan O Hodas. Few-shot learning with metric-agnostic conditional embeddings. *arXiv preprint arXiv:1802.04376*, 2018.

[16] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *CVPR*, 2017.

[17] Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Label propagation for deep semi-supervised learning. In *CVPR*, 2019.

[18] Thorsten Joachims. Transductive inference for text classification using support vector machines. In *ICML*, 1999.

[19] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012.

[21] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *ICLR*, 2017.

[22] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICML workshops*, 2013.

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

[24] Christiane Lemke, Marcin Budka, and Bogdan Gabrys. Metalearning: a survey of trends and technologies. *Artificial intelligence review*, 2015.

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

[26] Yu-Feng Li and Zhi-Hua Zhou. Towards making unlabeled data never hurt. *TPAMI*, 2014.

[27] Zhenguo Li, Fengwei Zhou, Fei Chen, and Hang Li. Meta-sgd: Learning to learn quickly for few-shot learning. *arXiv preprint arXiv:1707.09835*, 2017.

[28] Yanbin Liu, Juho Lee, Minseop Park, Saehoon Kim, Eunho Yang, Sung Ju Hwang, and Yi Yang. Learning to propagate labels: Transductive propagation network for few-shot learning. *arXiv preprint arXiv:1805.10002*, 2018.

[29] Takeru Miayto, Andrew M Dai, and Ian Goodfellow. Virtual adversarial training for semi-supervised text classification. 2016.

[30] Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. A simple neural attentive meta-learner. *arXiv preprint arXiv:1707.03141*, 2017.

[31] Tsendsuren Munkhdalai, Xingdi Yuan, Soroush Mehri, and Adam Trischler. Rapid adaptation with conditionally shifted neurons. *arXiv preprint arXiv:1712.09926*, 2017.

[32] Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018.

[33] Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. In *NeurIPS*, 2018.

[34] Limeng Qiao, Yemin Shi, Jia Li, Yaowei Wang, Tiejun Huang, and Yonghong Tian. Transductive episodic-wise adaptive metric for few-shot learning. In *ICCV*, 2019.

[35] Rajat Raina, Alexis Battle, Honglak Lee, Benjamin Packer, and Andrew Y. Ng. Self-taught learning: Transfer learning from unlabeled data. In *ICML*, 2007.

[36] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017.

[37] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. *arXiv preprint arXiv:1803.00676*, 2018.

[38] Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *science*, 2000.

[39] Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. *arXiv preprint arXiv:1807.05960*, 2018.

[40] Weiwei Shi, Yihong Gong, Chris Ding, Zhiheng MaXiaoyu Tao, and Nanning Zheng. Transductive semi-supervised deep learning using min-max features. In *ECCV*, 2018.

[41] Noah Simon, Jerome Friedman, and Trevor Hastie. A blockwise descent algorithm for group-penalized multiresponse and multinomial regression. *arXiv preprint arXiv:1311.6529*, 2013.

[42] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[43] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *NeurIPS*, 2017.

[44] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, 2014.

[45] Qianru Sun, Xinzhe Li, Yaoyao Liu, Shibao Zheng, Tat-Seng Chua, and Bernt Schiele. Learning to self-train for semi-supervised few-shot classification. *arXiv preprint arXiv:1906.00562*, 2019.

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

[47] Flood Sung, Li Zhang, Tao Xiang, Timothy Hospedales, and Yongxin Yang. Learning to learn: Meta-critic networks for sample efficient learning. *arXiv preprint arXiv:1706.09529*, 2017.

[48] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, 2017.

[49] Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 2000.

[50] Michael E Tipping and Christopher M Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 1999.

[51] Eleni Triantafillou, Richard Zemel, and Raquel Urtasun. Few-shot learning through an information retrieval lens. In *NeurIPS*, 2017.

[52] Vladimir Vapnik and Vlamimir Vapnik. Statistical learning theory wiley. *New York*, 1998.

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

[54] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical report, California Institute of Technology, 2011.

[55] Qizhe Xie, Eduard Hovy, Minh-Thang Luong, and Quoc V. Le. Self-training with noisy student improves imagenet classification. *arXiv preprint arXiv:1911.04252*, 2019.

[56] Sung Whan Yoon, Jun Seo, and Jaekyun Moon. Tapnet: Neural network augmented with task-adaptive projection for few-shot learning. *arXiv preprint arXiv:1905.06549*, 2019.

[57] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *NeurIPS*, 2014.

[58] Li Zhang, Tao Xiang, and Shaogang Gong. Learning a deep embedding model for zero-shot learning. In *CVPR*, 2017.

[59] Zhenyue Zhang and Hongyuan Zha. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM journal on scientific computing*, 2004.