On the Utility of Active Instance Selection for Few-Shot Learning

Pouya Pezeshkpour
University of California
Irvine, CA
pezeshkp@uci.edu

Zhengli Zhao
University of California
Irvine, CA
zhengliz@uci.edu

Sameer Singh
University of California
Irvine, CA
sameer@uci.edu

Abstract

Few-shot learning aims to understand new concepts by transferring knowledge and utilizing very few randomly selected labeled samples. Instead of selecting these instances randomly, active learning provides a promising alternative. In this work, we investigate the effectiveness of actively identifying informative samples on the performance of few-shot learning models. We show that despite the fact that regular classification tasks with larger amounts of labeled data benefit from active learning approaches, these benefits do not reliably generalize to the few-shot learning task. We characterize the best possible active few-shot learning performance, by introducing Single-Instance-Oracle and Batch-Oracle as active methods that assume access to labels of the unlabeled pool and the test set, and show via these "upper bounds" that we do not have a significant room for improving few-shot models through actively selecting instances.

1 Introduction

Deep learning models are accurate when large labeled datasets are available for training, however, adapting them to novel tasks requires effort in collecting labels; a significant problem in domains where data are expensive to label. *Few-shot learning* [20, 25] addresses this challenge by transferring the knowledge and reasoning learned from relevant datasets to the target evaluation dataset via a few shots of support instances. The goal of few-shot learning is to adapt models so that they can classify samples from unseen classes during training time, provided a small number of labeled instances from those classes [16, 5, 17, 2]. Since the number of labeled instances from these unseen classes is small, the role of an individual labeled instance dramatically increases compared to the usual supervised learning setting. Although previous approaches demonstrate tremendous success in transferring the knowledge, they disregard the possible impact of cautiously selecting those very few support instances. The question that remains is whether it is possible to improve few-shot models performance with fewer support instances if they are selected carefully.

Active learning (AL) [19, 7, 8] as an approach to iteratively select the *most informative* instances for the model to continuously train on, improving accuracy with a very few labeled instances, shares the same goal as few-shot learning, i.e., adopting models to a new task with a few samples. In spite of the attention AL methods have received, they mostly provide modest benefits for classification tasks [4, 13]. Our goal here is to explore the utility of actively selecting support instances on few-shot learning across models and datasets by studying current practices and characterizing the possible extent to which active learning can help few-shot models.

In this paper, we study the effect of combining active learning algorithms with few-shot learning by focusing on *active few-shot learning*. Specifically, we are interested in studying the effect of actively selecting the support set instances during the inference time of few-shot learning. Considering several existing active learning approaches, such as uncertainty based and Coreset methods [18], we evaluate the performance of two recent few-shot learning models, Prototypical Networks [20] and Simple

CNAPS [2] on in-domain and out-of-domain scenarios, respectively. Further, to characterize the possible benefit of active learning impact on few-shot models, assuming full access to labels of the unlabeled pool and the test set, we introduce *Single-Instance-Oracle* and *Batch-Oracle* active learning methods, by identifying instances that maximize the test set accuracy of the model when added to the support set. Since we only focus on the impact of active learning on few-shot models during the inference time, we can calculate the impact of selecting a new instance on the test set accuracy by performing a forward pass; in contrast to regular classification that requires retraining.

We evaluate the performance of active few-shot learning on CIFAR10, CIFAR100, and SVHN datasets, demonstrating the inability of current active learning techniques to effectively and consistently improve few-shot model performance. Further, upon achieving the upper bounds of the best possible performance for active few-shot learning task through our Single-Instance-Oracle, we show, (1) the room for improvement on active few-shot learning, depending on the benchmark and the model, may not be substantial considering the overall performance and number of labeled samples, and (2) more accurate few-shot learning models appear to benefit less from actively selecting instances. Finally, studying active few-shot learning when multiple instances are selected, by introducing Batch-Oracle, we shed some light on possible future directions.

2 Problem Setup

In this section, we introduce active and few-shot learning, setting up notations and relevant background for the remaining of the paper.

Few-Shot Learning In standard few-shot learning, we assume we have a large collection of instances $D = \{(x_i, y_i)\}$. From this dataset, we build separate *classification tasks* $D_T \subset D$ by randomly selecting data points without replacement. Each D_T (episode) consists of two separate sets, support set S_T and query set Q_T in a way that query set labels are a subset of support set labels. As a result, we can frame few-shot learning for each D_T as correctly assigning each data point x_i in the query set Q_T to one of the classes appears in S_T. Existing few-shot learning methods mostly consist of two phases, a training and a testing phase by assigning separate sets of D_T to each one of them in a way that test classes never appear in the training phase. In this work, we focus on Prototypical Networks [20] and Simple CNAPS [2] for in-domain and out-of-domain few-shot scenarios. We leave the family of few-shot models that use nearest neighbors [24, 9] to future work.

Active Learning In active learning, we are interested in learning a classifier that predicts a class from a set of labels $y \in C$ for any an input $x \subseteq \mathcal{R}^d$, trained on a large set of data point pairs (x_i, y_i). In this paper, we restrict our classifier to be a neural network represented as $\hat{y}_i = \sigma(\phi(x_i, \theta))$ for input instance x_i, where θ are parameters of the whole model, σ the softmax/sigmoid function, and the logits $\phi(x_i, \theta)$ come from a linear layer, i.e. $\phi(x_i, \theta) = \theta_l f_i$; θ_l are the parameters of final linear layer and f_i is a feature vector extracted from x_i. At each iteration t, active learning method has access to the current model (θ_t), a set of labeled instances S_t (contains pairs of x and y), and a large pool of unlabeled instances U_t (containing only x). In the batch setting, given a budget b, the goal is to identify b instances from the pool U that our active learning approach estimates, when they are labeled by an oracle, to be the *most informative* instances for the current classifier. Then, after querying the oracle for the labels and adding them to S_t, we retrain the classifier on the new training data, and continue the next iteration. Although in this work we focus on several uncertainty based and Coreset [18] AL methods, there are many other AL approaches that utlize the notion of uncertainty [21, 23, 18, 1, 27], identify instances that are not distinguishable from the pool of unlabeled samples by treating AL as a binary classifier [10] and incorporate the gradient of model [1].

Active Few-Shot Learning We introduce active few-shot learning as *actively* seeking the most informative unlabeled instance to add to the support set during the testing phase. The training phase is identical to the conventional few-shot learning task. We use five episodes in our setup during the testing phase, and use the entire test set for each class as the query set. Randomly choosing the classes to appear in these episodes, we gather all instances with those labels in the data and divide them equally into two sets: the pool that we choose the support images actively from, denoted as U_T, and the pool that we consider as the query set in the testing phase (Q_T). Our goal is to build S_T from U_T, in the testing phase, by choosing the most informative samples given the learned few-shot model. Although, to the best of our knowledge, we are the first to introduce active few-shot learning as described, the notion of active few-shot learning has been used in a few previous works. Woodward and Finn [26] combines reinforcement learning with one-shot learning to identify potentially informative samples to label. Moreover, the authors in [3] utilizes K-means clustering on the Prototypical Networks to better adapt few-shot models to new tasks. Finally, [6] tries to resolve task ambiguity when dealing with small amounts of data in few-shot learning by adopting MAML [5] model and injecting noise into gradient descent at meta-test time to develop a probabilistic meta-learning approach.

3 Active and Few-shot Learning models

Here we elaborate on different active and few-shot learning methods that we consider in this work.

3.1 Few-Shot Learning Models

To study the effect of active learning on the performance of few-shot models, we consider two models: 1) Prototypical Networks [20] because of its simplicity and popularity, and 2) Simple CNAPS [2] because of its state-of-the-art results for two separate few-shot scenarios, in-domain and out-of-domain respectively. In-domain represents scenarios in which we train and test a few-shot model on the same dataset dividing classes for training and testing purposes. On the other hand, out-of-domain represents scenarios in which we consider separate datasets for training and testing respectively. This methodology of evaluating a few-shot model becomes more widespread upon creation of the Meta-Dataset [22] which is a benchmark containing 10 different labeled image datasets.

Prototypical Networks Prototypical networks [20] classify each sample after calculating its feature vector through two steps: 1) computing the mean embedding or prototype for each class by averaging the embeddings of the samples of that class in the support set S_T, and 2) classifying the samples in the query set Q_T by calculating the similarity between each prototype and query samples' embedding. The similarity here is defined as a negative of Euclidean distance.

Simple CNAPS Simple CNAPS [2] is built on top of CNAPS [17] which uses FiLM layers [15] to better adapt the representations to unseen classes during the testing phase. More specifically, CNAPS classify the samples by firstly extracting the feature vector using ResNet18 [11] and FiLM layers, and then calculating the class probability using a linear classifier. The only difference between CNAPS and Simple CNAPS is incorporating the Mahalanobis distance instead of the linear classifier element.

3.2 Active Learning Models

Random: is a baseline that samples instances randomly in each iteration.

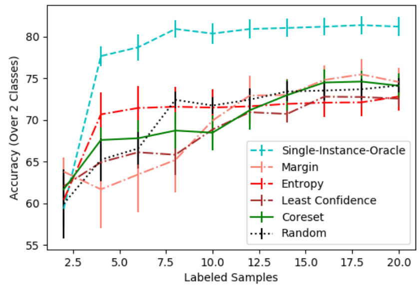
Uncertainty Based Sampling: We consider 3 uncertainty based approaches in this work: 1) **entropy** approach selecting samples by maximizing the entropy of class distributions based on classifier output for each unlabeled sample, 2) **least confidence** identifying instances by minimizing the highest probability class for each unlabeled sample, and 3) **margin of confidence** which chooses the instances by minimizing the difference between first and second highest class possibility.

Coreset [18]: selecting instances by using their representation from the last layer in DNN to solve the k-center problem, i.e., choosing b (our budget) centers in such a way that minimized the largest distance between a data point and its nearest center.

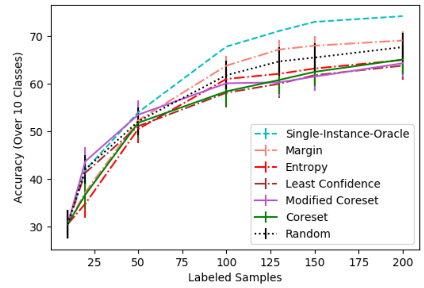
Modified Coreset: Considering that simple CNAPS uses Mahalanobis distance to classify instances, we introduce the *Modified Coreset* using Mahalanobis distance instead of Euclidean as a measure of distance in Coreset approach.

3.3 Oracle Based Active learning

In this section, to characterize the possible extent of active learning impact on the few-shot model, we first introduce an oracle based single instance selection approach. Then, conjecturing that single selection might impose a considerable limitation on the achievable upper bound, we provide a batch version of this method.



(a) Prototypical Networks.



(b) Simple CNAPS.

Figure 1: Active few-shot learning on **CIFAR-10**. Prototypical networks (in-domain) and Simple CNAPS models (out-of-domain) achieve 76.4 and 73.2 accuracy with all the labeled data.

Single-Instance-Oracle: To better grasp the possible effect of active learning on few-shot models, we introduce a Single-Instance-Oracle method: assuming to have full access to the label of all samples and the test set, we sample the most informative instance at each iteration by identifying the sample that maximizes the model accuracy on the test set upon adding to the support set.

Batch-Oracle: We define a batch version of our Single-Instance-Oracle, by keeping top K's most informative samples from the oracle algorithm perspective at each iteration in a beam. These K instances result in K different models (adding each one of the samples separately to the current support set). In the next iteration, we sort the informativeness of unlabeled instances according to these K models and then choose the top K most informative samples from the accumulated set.

Challenges and Utility: The biggest challenge in calculating Single-Instance-Oracle and Batch-Oracle is the computational complexity. Since at each iteration, we need to calculate the model accuracy on the test set upon adding each unlabeled sample separately, identifying the most informative samples even in a very limited label regime takes an order of magnitude longer than other methods. Just to note that, this approach is only applicable to few-shot learning since we only focus on the impact of active learning during the inference time. We can calculate the impact of selecting a new instance on the test set accuracy by having a forward pass of the learned model; in contrast to regular classification tasks which require to retrain the model for calculating this impact.

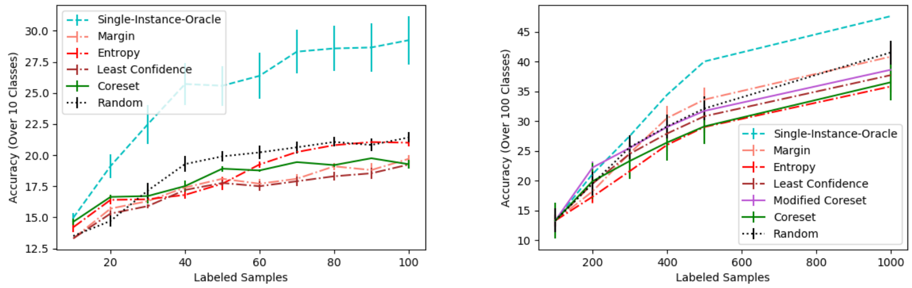
4 Experiments

Here, we first describe our benchmarks and implementation details. Then, we demonstrate the incapability of active learning approaches to consistently improve few-shot models in the in-domain setting. Finally, studying the out-of-domain scenario, we shed some light on future directions.

Setup To evaluate our method, we conduct several experiments on widely used image datasets CIFAR-10, CIFAR-100, SVHN. Both CIFAR-10 and CIFAR-100 [12] have $60K$ images of size 32×32 with 10 and 100 classes receptively. SVHN [14] contains 32×32 street view house number images with 10 labels. We trained all algorithms using the same optimization and loss, i.e., AdaGrad and the cross-entropy loss. We use validation data to tune the hyper-parameters and utilize a grid search to find the best hyper-parameters, such as learning rate, and regularization parameter. In all of our experiments, we report the resulted accuracy averaged over 10 runs with different random seed.

4.1 Impact of Active Selection on Few-Shot Learning

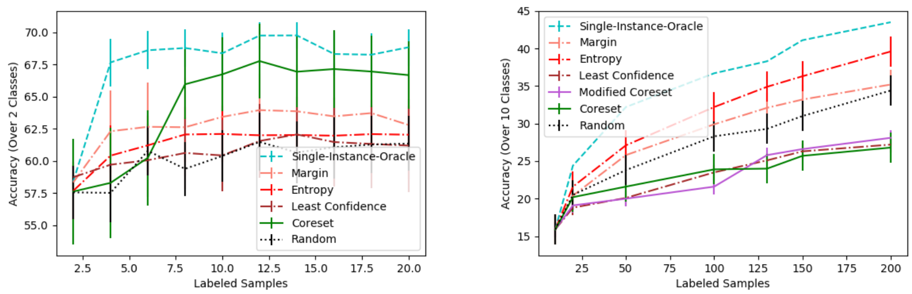
In this section we study the effect of active learning for two families of methods; 1) methods that sample instances independent of each other, i.e., estimating the effect of adding a single point to training data on the model (we consider the uncertainty based models as representative of this family), and 2) methods that adopt batch-wise strategies as their sampling procedure, i.e., selecting data points estimating the effect of a batch of samples together (we consider Coreset as an example of such).



(a) Prototypical Networks.

(b) Simple CNAPS.

Figure 2: Active few-shot learning on **CIFAR-100**. Prototypical networks (in-domain) and Simple CNAPS models (out-of-domain) achieve 27.7 and 47.4 accuracy using all the labeled data.



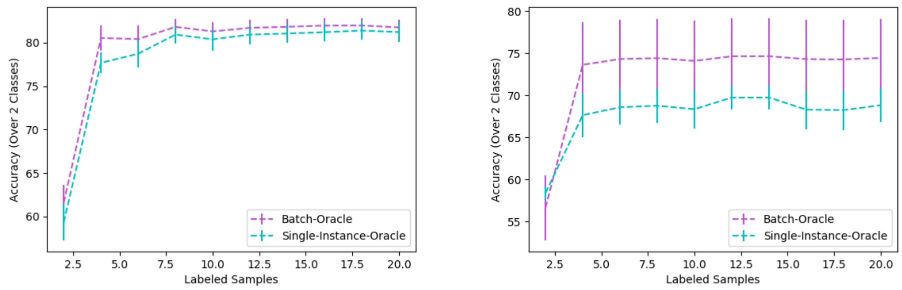
(a) Prototypical Networks.

(b) Simple CNAPS.

Figure 3: Active few-shot learning on **SVHN**. Prototypical networks (in-domain) and Simple CNAPS models (out-of-domain) achieve 71.1 and 44.2 accuracy using all the labeled data.

In-domain: We introduce active few-shot learning task, by actively choosing support images in the test phase. Specificity we define support and query images for different episodes during training using previously introduced instructions [24, 22]. For the test phase query images of unseen classes, we consider all images with these labels from the test data, i.e., test set of CIFAR-10, CIFAR-100, and SVHN datasets. As for sampling the classes for the test phase, since we only consider one episode during the test phase, for CIFAR-10 and SVHN we randomly choose two of the classes as the unseen ones, and for CIFAR-100 we randomly choose 10 classes (use the rest of classes for training the model). Moreover, for the test phase support images, we start with one random image for each label, and then increase the support images in each iteration using an active learning approach. We train CIFAR-10 and SVHN in 5 shot-5 way manner, while for the CIFAR-100 datasets we train few-shot model in a 5 shot-10 way setting. The average result of 10 runs over in-domain few-shot learning benchmarks is shown in Figures 1a, 2a and 3a. We can derive two conclusions from these results: 1) current active learning methods do not show sufficient improvements for few-shot learning, 2) even Single-Instance-Oracle does not consistently improve the performance of the model. Finally, the Prototypical networks using all possible labels as part of the support set achieves 76.4, 27.7, and 71.1 accuracy on CIFAR-10, CIFAR-100, and SVHN benchmarks which is very close to Single-Instance-Oracle end point accuracy in each plot.

Out-of-domain: As our few-shot learning model in the out-of-domain scenario we use pretrained weights of simple CNAPS [2] model. For the test phase query images, we adopt a similar approach to our previously described in-domain scenario with one major difference, i.e., considering all classes of CIFAR-10, CIAFR-100, and SVHN during the testing phase. The result of few-shot learning benchmarks in the out-of-domain scenario is shown in Figures 1b, 2b and 3b. These results further validate our previous conclusion on the incapability of current active learning models in improving the few-shot learning task. Considering the gap between Single-Instance-Oracle and random active sampling, it appears the more accurate few-shot models, (simple CNPAS comparing to Prototypical



(a) Prototypical Networks on CIFAR-10.

(b) Prototypical Networks on SVHN.

Figure 4: Results on active few-shot for Single-Instance-Oracle vs Batch-Oracle over CIFAR-10 and SVHN with Prototypical Networks as few-shot model. We consider beam size of 2 in this evaluation.

Networks), are less promising in active few-shot task. Moreover, the poor performance of Coreset and Modified Coreset demonstrates the inherently different nature of active few-shot learning task in comparison to regular active learning. Finally, the Simple CNAPS using all possible labels as part of the support set achieves 73.2, 47.4, and 44.2 accuracy on CIFAR-10, CIFAR-100, and SVHN benchmarks which is very close to Single-Instance-Oracle end point accuracy in each plot.

Although current active learning methods perform poorly in both in-domain and out-of-domain scenarios, batch-wise strategies indicates there is some room for improvement (Coreset on in-domain SVHN, and Modified Coreset on out-of-domain CIFAR-10 and CIFAR-100, especially when dealing with fewer samples). A natural question that arises is whether we can quantify how much more can picking multiple instances in a batch can benefit, over the Single-Instance-Oracle.

4.2 Effect of Batch-wise Sampling on Active Few-Shot

In this section, we compare our Single-Instance-Oracle and Batch-Oracle methods, to clarify if it is possible to further improve model performance on introducing better batch-wise strategies. We only consider small batches ($K = 2$) because of the excessive computational cost in Batch-Oracle setting and evaluate our methods for the in-domain CIFAR-10 and SVHN benchmarks. The average result of Batch-Oracle performance is provided in Figure 4. As it shows, incorporating a beam search on top of the Single-Instance-Oracle method improves the model performance, showing room for improvement of active few-shot methods by adopting better batch-wise active learning algorithms.

4.3 Relation between Oracle Selection and Uncertainty Measures

To understand to what degree the uncertainty measures reflect the points selected by the Oracle, we compute the correlation between them on the ranking over the unlabeled samples. For this evaluation, we start with 5 random samples for each class in the support set and then rank the unlabeled samples based on each method during the inference time. Spearman correlation coefficients between oracle selection and uncertainty measures are provided in Table 1. For both Simple CNAPS and Prototypical networks models, uncertainty measures do not seem to be correlated with Oracle selection, further emphasizing the need for introducing an active learning approach that goes beyond the notion of uncertainty. Moreover, Prototypical networks show a weaker correlation, which is consistent with the bigger accuracy gap we observe between uncertainty methods and Single-Instance-Oracle selection, compared to Simple CNAPS.

5 Conclusions

Motivated by the need for actively selecting support set images in few-shot learning, we study the effect of active learning on few-shot models. Establishing the upper bounds of the best possible active few-shot learning performance by introducing Single-Instance-Oracle, with the assumption of full access to the test set and the labels of unlabeled samples pool, we show: 1) current active learning methods cannot improve few-shot models substantially in comparison to random sampling,

Table 1: Spearman correlation coefficient between oracle selection and uncertainty methods over unlabeled samples in CIFAR-10 dataset. We randomly choose 5 samples for each class in the support set and compute the scores of remaining unlabeled instances based on our active method.

Method	**Spearman's ρ**	
	Prototypical	Simple CNAPS
Entropy	0.028	0.277
Margin of Confidence	0.023	0.269
Least Confidence	0.024	0.293

2) even Single-Instance-Oracle method fall short in improving the few-shot models' performance with a considerable gap. Further, studying the effect of batch-wise active sampling by introducing Batch-Oracle, we concluded a possible room for improvement upon introducing more effective batch-wise selection.

In future work, for active learning to substantially improve upon simple baselines like random and uncertainty sampling, it is essential to introduce new *batch-wise* active learning methods that utilize the unique characteristics of the few-shot learning tasks. Although in this work we define active few-shot learning as actively selecting support set instances during the testing phase, other varieties of this problem, such as actively identifying informative instances for support and query set during the training time, remain open questions to pursue in future work.

Acknowledgements

We would like to thank the anonymous reviewers for their detailed feedback and suggestions. This work is sponsored in part by the Air Force Research Laboratory (AFRL), and DARPA, and funded in part by the NSF award #IIS-1756023. The views expressed are those of the authors and do not reflect the official policy or position of the funding agencies.

References

[1] J. T. Ash, C. Zhang, A. Krishnamurthy, J. Langford, and A. Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*, 2019.

[2] P. Bateni, R. Goyal, V. Masrani, F. Wood, and L. Sigal. Improved few-shot visual classification. *arXiv preprint arXiv:1912.03432*, 2019.

[3] R. Boney and A. Ilin. Semi-supervised and active few-shot learning with prototypical networks. *arXiv preprint arXiv:1711.10856*, 2017.

[4] L. P. Evans, N. M. Adams, and C. Anagnostopoulos. When does active learning work? In *International Symposium on Intelligent Data Analysis*, pages 174–185. Springer, 2013.

[5] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135. JMLR. org, 2017.

[6] C. Finn, K. Xu, and S. Levine. Probabilistic model-agnostic meta-learning. In *Advances in Neural Information Processing Systems*, pages 9516–9527, 2018.

[7] S. Freeman, S. L. Eddy, M. McDonough, M. K. Smith, N. Okoroafor, H. Jordt, and M. P. Wenderoth. Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences*, 111(23):8410–8415, 2014.

[8] Y. Gal, R. Islam, and Z. Ghahramani. Deep bayesian active learning with image data. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1183–1192. JMLR. org, 2017.

[9] V. Garcia and J. Bruna. Few-shot learning with graph neural networks. *arXiv preprint arXiv:1711.04043*, 2017.

[10] D. Gissin and S. Shalev-Shwartz. Discriminative active learning. *arXiv preprint arXiv:1907.06347*, 2019.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[12] A. Krizhevsky, G. Hinton, et al. Learning multiple layers of features from tiny images. *Citeseer*, 2009.

[13] D. Lowell, Z. C. Lipton, and B. C. Wallace. Practical obstacles to deploying active learning. *arXiv preprint arXiv:1807.04801*, 2018.

[14] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng. Reading digits in natural images with unsupervised feature learning. *NeurIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[15] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville. Film: Visual reasoning with a general conditioning layer. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

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

[17] J. Requeima, J. Gordon, J. Bronskill, S. Nowozin, and R. E. Turner. Fast and flexible multi-task classification using conditional neural adaptive processes. In *Advances in Neural Information Processing Systems*, pages 7957–7968, 2019.

[18] O. Sener and S. Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017.

[19] B. Settles. Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, 2009.

[20] J. Snell, K. Swersky, and R. Zemel. Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, pages 4077–4087, 2017.

[21] S. Tong and D. Koller. Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov):45–66, 2001.

[22] E. Triantafillou, T. Zhu, V. Dumoulin, P. Lamblin, U. Evci, K. Xu, R. Goroshin, C. Gelada, K. Swersky, P.-A. Manzagol, et al. Meta-dataset: A dataset of datasets for learning to learn from few examples. *arXiv preprint arXiv:1903.03096*, 2019.

[23] G. Tur, D. Hakkani-Tür, and R. E. Schapire. Combining active and semi-supervised learning for spoken language understanding. *Speech Communication*, 45(2):171–186, 2005.

[24] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al. Matching networks for one shot learning. In *Advances in neural information processing systems*, pages 3630–3638, 2016.

[25] Y. Wang, Q. Yao, J. Kwok, and L. Ni. Few-shot learning: A survey. *arXiv preprint arXiv:1904.05046*, 2019.

[26] M. Woodward and C. Finn. Active one-shot learning. *arXiv preprint arXiv:1702.06559*, 2017.

[27] F. Zhdanov. Diverse mini-batch active learning. *arXiv preprint arXiv:1901.05954*, 2019.