Exploring Representation Learning for Flexible Few-Shot Tasks

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

Existing approaches to few-shot learning deal with tasks that have persistent, rigid notions of classes. Typically, the learner observes data only from a fixed number of classes at training time and is asked to generalize to a new set of classes at test time. Two examples from the same class would always be assigned the same labels in any episode. In this work, we consider a realistic setting where the relationship between examples can change from episode to episode depending on the task context, which is not given to the learner. We define two new benchmark datasets for this flexible few-shot scenario, where the tasks are based on images of faces (Celeb-A) and shoes (Zappos50K). While classification baselines learn representations that work well for standard few-shot learning, they suffer in our flexible tasks since the classification criteria shift from training to testing. On the other hand, unsupervised contrastive representation learning with instance-based invariance objectives preserves such flexibility. A combination of instance and class invariance learning objectives is found to perform best on our new flexible few-shot learning benchmarks, and a novel variant of Prototypical Networks is proposed for selecting useful feature dimensions.

1 Introduction

Following the success of machine learning applied to fully-supervised settings, there has been a surge of interest in machine learning within more realistic, natural learning scenarios. Among these, meta-learning and few-shot learning [11] (FSL) have emerged as exciting alternatives. In the few-shot learning setting, the learner is presented with episodes of new learning tasks, where the learner must identify patterns in a labeled support set and apply them to make predictions for an unlabeled query set. Since its inception, there has been significant progress on FSL benchmarks. However, standard supervised baselines are often shown to perform as well as carefully designed solutions [3, 16]. In this work, we argue that this observation is due in part to the rigidity in which FSL episodes are designed.

In a typical few-shot classification setting, each episode consists of a few examples belonging to one of N classes. Across different training episodes, different images are sampled from the classes in the training set but they will always be given the same class label: an elephant is always an elephant. Most current approaches to FSL attempt to remove context. Existing tasks focus on classification judgements, where the query image should be deemed similar to the support image belonging to the same class, factoring out the role of context such as the setting, pose, and presence of other objects. But many judgements are contextual—they depend on the task at hand and frame-of-reference. A rock is similar to a chair when the aim is to sit, but similar to a club if the aim is to hit. Meta-learning is especially appropriate in contextual judgements, as people are able to adapt readily to new contexts and make appropriate judgements. So an important question is how to get context into few-shot classification?

Submitted to the 4th Workshop on Meta-Learning at NeurIPS 2020, Vancouver, Canada. Do not distribute.

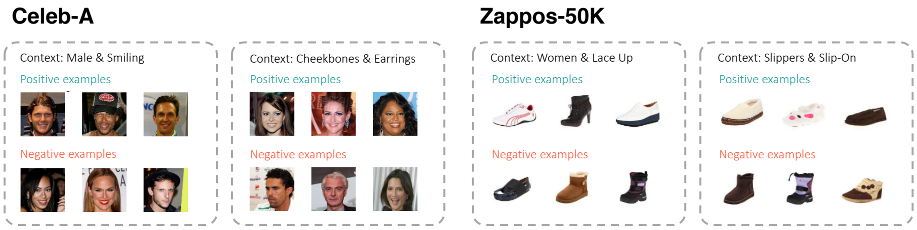

Figure 1: Sample FFSL episodes using Celeb-A (left) and Zappos-50K (right) datasets. Positive and negative examples are sampled according to the context attributes, but the context information is not revealed to the model at test time.

In this work, we define a new flexible few-shot learning (FFSL) paradigm. Instead of building episodes from classes, each episode is a binary classification problem that is constructed with some context that is hidden from the learner. In this way, the same data point may be given different labels across multiple episodes. For example, elephants and tables may belong to the same class if the context is "has legs", but not when the context is "has ears". Importantly, the learner is not given direct access to the context and must infer it from the examples present in the episode.

Our FFSL problem is significantly more challenging than the standard setup. In each episode, a learner must infer the correct context and adapt their predictions accordingly. In Section 3.1 we study generalization issues that occur under supervised representation learning for the flexible few-shot tasks. We show that these approaches easily overfit to the training attributes, even when given direct access to the attributes that determine the context. We provide additional analysis of a toy problem to illustrate one possible cause of this failure.

In this work, we contribute two new benchmark datasets for this flexible few-shot scenario. The tasks are based on images of faces (Celeb-A) [12] and shoes (Zappos50K) [20]. We provide a thorough empirical evaluation of existing methods on these tasks. We find that successful approaches in the standard FSL setting fall short on the flexible few-shot tasks. Further, while supervised classification baselines can learn good representation in the standard FSL setting, they suffer in FFSL. On the other hand, we found a combination of instance and class invariance objectives is able to provide improved performance on the flexible few-shot tasks. Moreover, we present Mask-ProtoNet which combines prototype classification with feature selection capability, and it performs better compared to standard prototype averaging and linear readout.

2 FFSL: Flexible Few-Shot Learning

In this section, we define our FFSL paradigm and introduce our two new benchmark datasets. As in the standard few-shot classification setting (Section **??**), our learner is presented with episodes of data. However, the episodes are not constrained to contain data points from only N classes. Instead, each data point is given either a positive or negative label depending on some criteria that is not known to the learner.

Conceptually, each data point $\mathbf{x} \in \mathcal{X}$ represents some combination of hidden attributes $\mathbf{z} \in \mathcal{Z}$. And each context is an injective function, $f : \mathcal{Z} \rightarrow \{0, 1\}$, that labels each of the data points depending on their hidden attributes. In this work, we consider contexts that compute conjuctions of binary attributes. The set of training contexts and test contexts need not be the same.

In order to solve the FFSL task, the learner must correctly find a mapping from the data domain $\mathcal{X}$ to the correct labels. One natural way to solve this problem would be to first find a mapping $h : \mathcal{X} \rightarrow \mathcal{Z}$, that is persistent across episodes, and then estimate the context in each episode. However, we do not limit our exploration to methods that use this approach, since FFSL allows different partitions of the $\mathcal{Z}$ space for training and testing, and as we will explain in Section 3.1, directly learning to predict $\mathcal{Z}$ can lead to generalization issues.

Next we describe how we generate the FFSL datasets using existing image datasets with attributes. Sample episodes from each dataset are shown in Figure 1.

Celeb-A: The Celeb-A dataset [12] contains around 200K images, where we split half to training, and a quarter to validation and testing each. Each image is annotated with 40 binary attributes, detailing hair colour, facial expressions, and other descriptors. We picked 27 salient attributes and split 14 for training and 13 for both val and test. There is no overlap between training or test attributes but they may sometimes belong to a common category, e.g. blonde hair is in training and brown hair is in test. For each episode, we randomly select one or two attributes and look for positive example belonging to these attributes simultaneously. And we also sample an equal number of negative examples that don't belong to one or both of the selected attributes. This will construct a *support set* of positive and negative samples, and then we repeat the same process for the corresponding *query set* as well.

Zappos-50K: The Zappos-50K dataset [20] contains just under 50K images of shoes annotated with attribute values, out of which we kept a total of 76 that we considered salient. We construct an image-level split that assigns 80% of the images to the training set, 10% to the validation and 10% to the test set. We additionally split the set of attribute values into two disjoint sets that are used to form the training and held-out FFSL tasks, respectively. Sampling an episode from a particular split involves sampling a conjunction of attributes from that split (e.g. 'gender = boy' and 'material = leather'), and then sampling positive and negative examples from the relevant example split. The positive examples obey both clauses of the conjunction and, as a design choice, the negative examples do not obey either clause. The sampled positive and negative examples are then divided into a support and query set for the episode.

3 Exploring Models for Flexible Few-Shot Learning

In this section, we explore different learning models to solve FFSL tasks. Overall, we separate learning into two stages: *representation learning* and *few-shot learning*. In the representation learning stage, a network backbone learns task relevant features over many examples. And in the FSL stage, an episode with a few examples is presented, and the learner utilizes the base backbone network and performs additional learning on top.

For typical *meta-learning* based methods, these two stages are essentially the same—training performs episodic learning just like testing. Aside from meta-learning, simple supervised pretraining can also learn good representation for standard few-shot classification by using a linear classifier readout at test time [3, 16].

3.1 Generalization issues with supervised representation learning

In the FFSL task, any single example can have several positive attributes and the context used to classify them varies across training and test. This suggests that useful representations must be more general than those needed for standard FSL. To investigate this, we first conducted an initial experiment on the Celeb-A benchmark. We adopted a standard prototypical network (**ProtoNet**) with features learned through the episodic query loss as our meta-learning approach. We also explored pretraining-based approaches. We trained a classifier to predict the 14 binary training attributes from the input images to learn a representation. At test time we simply used a linear classifier to solve each episode. This approach is denoted as **SA** (Supervised Attributes), analogous to the setting in [3]. We also trained an oracle classifier (**SA***) on all 40 attributes in the dataset, including both training and testing attributes. Since the tasks are constructed using attribute information, the performance of **SA*** should be considered an upper bound for this problem.

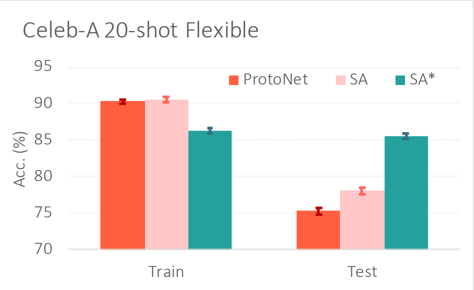

Figure 2: FFSL 20-shot classification. Both supervised attribute classification and standard FSL do not generalize well.

Results are shown in Figure 2. Both ProtoNet and SA perform well on the training tasks since they are exposed to the label information from the training attributes; however, the test performance shows a significant generalization gap. In order to succeed in the training objective, both ProtoNet and SA essentially learn to ignore other features that are potentially useful for testing as classification criteria. By contrast, SA* is able to perform similarly on both training and testing, since the learning does not depend on a particular split of the attributes. Initial experiments therefore suggest that supervised learning alone will likely not be sufficient for our FFSL task.

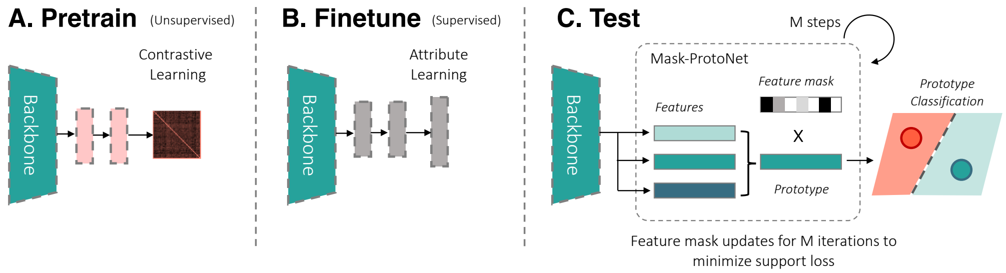

Figure 3: **Our proposed method for FFSL. A:** we first pretrain the network with unsupervised contrastive objective to learn general features. **B:** Then we finetune the network to classify the set of training attributes. Both stages employ a different decoder header so that the representation remains general. **C:** Finally at test time we use Mask-ProtoNet, a variant of ProtoNet that infers feature selection iteratively.

In Appendix we study a toy FFSL problem which further illustrates these generalization issues. We explore training a prototypical network on data from a linear generative model, where each episode presents significant ambiguity in resolving the correct context. We show that in this setting, unlike in standarad FSC tasks, the prototypical network is forced to discard information on the test attributes in order to solve the training tasks effectively, and thus fails to generalize.

3.2 Unsupervised constrastive representation learning

Learning good representation for downstream applications has always been a sought-after purpose of deep learning. [7] proposed to pretrain subsequent layers of autoencoders for representation learning, and showed good performance for dimensionality reduction, and downstream classification. Following the development of variational autoencoders (VAEs) [9], many extensions have been proposed to encourage "disentangled" representation learning by reweighing terms in the evidence lower bound [6, 8].

In contrast to traditional generative modeling where the objective is grounded on uncovering the data distribution, self-supervised learning recently emerged as a promising approach for representation learning. These include learning to predict rotations [10], maximize mutual information between the input and representation [1, 17], and contrastive learning approaches [2, 17, 15, 4, 18]. They have shown promise in learning semantic aware representations, almost closing the gap with supervised representation training on the challenging ImageNet benchmark. We follow SIMCLR [2] as a representative framework for unsupervised contrastive learning, shown in Figure 3-A. We chose SIMCLR because of its empirical success.

Concretely, it sends a pair of augmented versions of the same image to the input and obtains a hidden representation. The hidden representation is further passed into a decoder, producing unit-norm vectors. The network is trained end-to-end to minimize the InfoNCE loss [17], which distinguishes the positive sample from the same pair from the rest by encouraging feature dot product between the positive pair to gain a higher value than negative pairs.

Finetuning with supervised attribute classification We can combine the merits of unsupervised representation learning and supervised attribute classification (SA). To prevent SA from overriding the unsupervised features, we add another classifier decoder MLP before the sigmoid classification layer (see Figure 3-B). Empirically, finetuning on SA is found to be beneficial, but early stopping is needed to prevent optimizing too much towards training attributes, which would cause significant generalization issues (Section 3.1).

During test time, we directly use the representation before both decoders to perform FSL. In the next section, we introduce Mask-ProtoNet, a novel method for FFSL.

3.3 Few-Shot Learning with Mask-ProtoNet

Once the representation is learned, a common approach for FSL is to directly learn a linear classifier on top of the representation, or average the prototypes from the support set. Prototype averaging, however, will consider all feature dimensions, including the ones that are not relevant to the current episode. A linear classifier, on the other hand, learns a weight coefficient for each feature dimension, thus performing some level of feature selection. Still, the weights need to be properly regularized

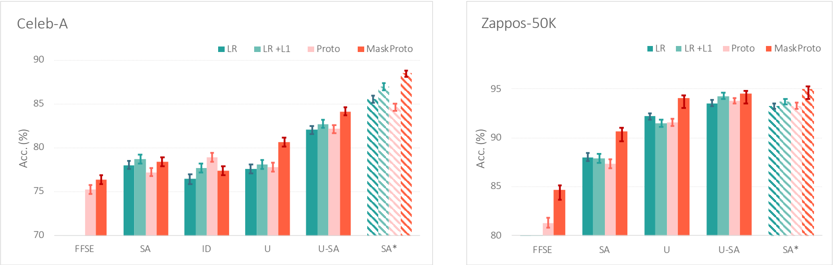

Figure 4: **20-shot FFSL results comparing different representation learning and FSL stage combinations. FFSE**: Meta-learning directly using the flexible few-shot episodes. **SA**: Supervised attribute classification. **ID**: Auxiliary representation learning (for Celeb-A this is face ID classification). **U**: Unsupervised contrastive learning. **U-SA**: Our proposed U pretraining followed by SA finetuning. **SA***: Supervised attribute binary classification on **all** attributes, which serves as an oracle (striped bars). A set of few-shot learners are evaluated: 1) logistic regression (**LR**), 2) LR with L1 regularization (**LR +L1**), 3) ProtoNet (**Proto**), and 4) the proposed Mask-ProtoNet (**MaskProto**). U-SA with Mask-ProtoNet achieves the best performance in both benchmarks. Chance is 50%.

to encourage high-fidelity selection. A popular way is to apply an L1 regularizer on the weights to encourage sparsity. The learning of a classifier is essentially done at the same time as the selection of feature dimensions. In this paper, we propose Mask-ProtoNet as an alternative for few-shot learning that separates the procedure of classifier learning and feature selection: we use prototypes for classification and additionally learn a soft binary mask for feature selection.

Just like a linear classifier, the Mask-ProtoNet learns a weight coefficient for each dimension. This weight is then passed through a sigmoid function to act as a soft binary mask, which is learned for a small number of iterations before termination. Finally classification is performed based on the masked prototypes. Conceptually, the mask will disable unused features and instead focus on dimensions that are activated in the current episode. The mask is updated to minimize the inner loop loss, which is a combination of support set cross entropy and an L1 sparse regularizer. The full algorithm is described in Algorithm 1 and Figure 3-C.

Algorithm 1 Mask-ProtoNet

```
Require: Net, $\{\mathbf{x}_i^S, y_i^S\}_{i=1}^N$, $\{\mathbf{x}^Q\}_{j=1}^M$
    // An embedding network, $N$ support, $M$ query
Ensure: $\{\hat{y}_j^Q\}_{j=1}^M$
    // Network representation $\mathbf{h} \in \mathbb{R}^D$
$\mathbf{h}_i^S \leftarrow \text{Net}(\mathbf{x}_i^S)\ \ \forall i;\ \ \mathbf{h}_j^Q \leftarrow \text{Net}(\mathbf{x}_j^Q)\ \ \forall j;$
$\mathbf{w} \leftarrow 0 \in \mathbb{R}^D;$
for $t = 1 \ldots M+1$ do
    $\tilde{\mathbf{w}} \leftarrow \sigma(\mathbf{w})$
    $\mathbf{p}[k] \leftarrow \frac{\sum_i (\mathbf{h}_i^S \odot \tilde{\mathbf{w}}) \mathbb{1}[y_i^S = k]}{\sum_i \mathbb{1}[y_i^S = k]}$
    $\hat{y}_{i,k}^S \leftarrow \text{softmax}(-d(\mathbf{h}_i^S \odot \tilde{\mathbf{w}}, \mathbf{p}[k]))\ \ \forall i;$
    $l \leftarrow -\frac{1}{N} \sum_i CE(\hat{y}_i^S, y_i^S) + \lambda \|\tilde{\mathbf{w}}\|_1$
    $\mathbf{w} \leftarrow \mathbf{w} - \eta \nabla_{\mathbf{w}} l$
end for
$\hat{y}_{j,k}^Q \leftarrow \text{softmax}(-d(\mathbf{h}_j^Q \odot \tilde{\mathbf{w}}, \mathbf{p}[k]))\ \ \forall j;$
return $\hat{y}_j^Q$
```

4 Experiments

Implementation details: Images were resized to 84 × 84 × 3. We used ResNet-12 [5, 14] with 64, 128, 256, 512 channels in each residual module. The decoder network for contrastive learning has two 512-d layers and outputs 128-d vectors. The classifier finetuning decoder network has two 512-d layers and outputs a 512-d vector. We trained SIMCLR using random crop areas of 0.08 – 1.0, color augmentation 0.5, and InfoNCE temperature 0.5, for 1000 epochs using LARS [19] and cosine schedule with batch size 512 and peak learning rate 2.0. SA finetuning lasts for another 2k steps with batch size 128 and learning rate 0.1 for the decoder and 0.01 for the backbone and momentum 0.9. ID, SA and SA* use batch size 256 with a learning rate 0.1 for 30k steps, with 0.1x learning rate decay at 20k and 25k steps, and momentum 0.9. Features are normalized before sending to LR classifiers. We use cosine similarity for ProtoNet and Mask-ProtoNet.

4.1 Results and discussion

Main results: Figure 4 shows our main results on Celeb-A and Zappos-50K with 20-shot FFSL episodes. On both benchmarks, training on flexible few-shot episodes based on training attributes (FFSE) performed worst. This aligns with our observation of the generalization issue explained in Section 3.1. Similarly, supervised attribute (SA) learning faced the same challenge. An auxiliary task of facial identification (ID) was not helpful for representation learning either. Interestingly,

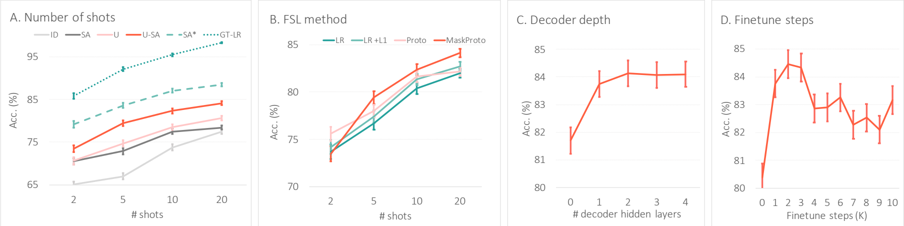

Figure 5: Additional results on the Celeb-A dataset. A: How many examples are needed for FFSL? B: Comparison of few-shot learning methods on different number of shots. C: Effect of the number of decoder layers during finetuning. D: Effect of the number of finetuning steps.

unsupervised representation learning (U) attained relatively better test performance, suggesting that the training objective in contrastive learning preserves more general features—not just shown for semantic classification tasks in prior literature, but also for the flexible class definitions present here. Surprisingly, finetuning slightly on SIMCLR pretrained networks (U-SA) contributed further gains in performance. We also tried to finetune directly on FFSL episodes using meta-learning approaches but this did not perform well — one possible explanation is given in our toy example in the Appendix. We conclude that meta-learning may not help learn higher-level features about the FFSL task itself. Lastly, we confirmed that U-SA closes the generalization gap between SA and SA*, and obtained matching performance on Zappos-50K. Lastly, we confirmed that U-SA closes the generalization gap between SA and SA*. These results were consistent across our benchmarks. Therefore, U-SA was the most effective representation learning algorithm we explored for FFSL. Note that this result contrasts with standard FSL literature, where unsupervised representation learning still lags behind supervised pretraining [13]. Moreover, MaskProto is often the best across different FSL approaches, consistently higher than Proto, which does not reason about feature selection.

Number of shots: Since we have a flexible definition of classes in each episode, it could be the case that the support examples are ambiguous. For example, by presenting both an elephant and a cat in the support set, it is unclear whether the positive set is about animals or mammals. Results are shown in Figure 5-A. In addition to the SA* oracle, we provided another oracle GT-LR, where the representations are the binary attribute values, and readout from a linear classifier. GT-LR gradually approached 100% accuracy as the number of shots approached 20. This demonstrates that FFSL tasks potentially require more support examples to resolve ambiguity. Again, U-SA consistently outperformed other baselines. Figure 5-B plots the performance of different FSL methods, using a common U-SA representation. Mask-ProtoNet performs better with more support examples, but worse with fewer (e.g. 2), since minimizing the loss of only 2 examples can lead to over-confidence.

Effect of decoder depth: Figure 5-C studies the effect of a decoder for attribute classification finetuning. Adding an MLP decoder was found to be beneficial for unsupervised representation learning in prior literature [2]. Here we found that adding a decoder is also important for SA finetuning, contributing to over 2% improvement.

Effect of SA finetuning: Figure 5-D plots the validation accuracy on FFSL tasks during finetuning for a total of 10k steps. It is found that the accuracy grows from 80% and peaks at 2k steps with over 84%, and then drops. This suggests that a little finetuning on supervised attributes is beneficial, but prolonged finetuning eventually makes the representation less generalizable.

5 Conclusion

The notion of a class often changes depending on the context, yet existing few-shot classification relies on a fixed semantic class definition. In this paper, we propose a flexible few-shot learning paradigm where the classification criteria change based on the episode context. We proposed benchmarks using the Celeb-A and Zappos-50K datasets to create flexible definitions with existing attribute labels. We explored various ways to perform representation learning for this new task. Unlike in standard FSL, we found that supervised representation learning generalizes poorly on the test set, due to the partitioning of training & test attributes. Unsupervised contrastive learning on the other hand preserved more generalizable features, and further finetuning on supervised attribute classification yielded the best results. Finally, a variant of ProtoNet, Mask-ProtoNet is proposed and delivers better readout performance. The development of FFSL benchmarks will hopefully encourage more future research investigating the generalization ability of meta-learning methods.

References

[1] M. I. Belghazi, A. Baratin, S. Rajeswar, S. Ozair, Y. Bengio, A. Courville, and R. D. Hjelm. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018.

[2] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton. A simple framework for contrastive learning of visual representations. *CoRR*, abs/2002.05709, 2020.

[3] W. Chen, Y. Liu, Z. Kira, Y. F. Wang, and J. Huang. A closer look at few-shot classification. In *Proceedings of the 7th International Conference on Learning Representations, ICLR*, 2019.

[4] K. He, H. Fan, Y. Wu, S. Xie, and R. B. Girshick. Momentum contrast for unsupervised visual representation learning. *CoRR*, abs/1911.05722, 2019.

[5] 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, CVPR*, 2016.

[6] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *5th International Conference on Learning Representations, ICLR*, 2017.

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

[8] H. Kim and A. Mnih. Disentangling by factorising. In J. G. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML*, 2018.

[9] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[10] A. Kolesnikov, X. Zhai, and L. Beyer. Revisiting self-supervised visual representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1920–1929, 2019.

[11] B. M. Lake, R. Salakhutdinov, J. Gross, and J. B. Tenenbaum. One shot learning of simple visual concepts. In *Proceedings of the 33th Annual Meeting of the Cognitive Science Society, CogSci*, 2011.

[12] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision, ICCV*, 2015.

[13] C. Medina, A. Devos, and M. Grossglauser. Self-supervised prototypical transfer learning for few-shot classification. *CoRR*, abs/2006.11325, 2020.

[14] B. N. Oreshkin, P. R. López, and A. Lacoste. TADAM: task dependent adaptive metric for improved few-shot learning. In S. Bengio, H. M. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31, NeurIPS*, 2018.

[15] Y. Tian, D. Krishnan, and P. Isola. Contrastive multiview coding. *CoRR*, abs/1906.05849, 2019.

[16] Y. Tian, Y. Wang, D. Krishnan, J. B. Tenenbaum, and P. Isola. Rethinking few-shot image classification: a good embedding is all you need? *CoRR*, abs/2003.11539, 2020.

[17] A. van den Oord, Y. Li, and O. Vinyals. Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748, 2018.

[18] Y. Xiong, M. Ren, and R. Urtasun. Loco: Local contrastive representation learning. *CoRR*, abs/2008.01342, 2020.

[19] Y. You, I. Gitman, and B. Ginsburg. Scaling SGD batch size to 32k for imagenet training. *CoRR*, abs/1708.03888, 2017.

[20] A. Yu and K. Grauman. Fine-grained visual comparisons with local learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, 2014.