Multi-label Iterated Learning for Image Classification with Label Ambiguity

Sai Rajeswar[1,2,3*], **Pau Rodríguez**[1*], **Soumye Singhal**[2,3], **David Vazquez**[1], **Aaron Courville**[2,3,4]
[1]ServiceNow Research, [2]Montréal Institute of Learning Algorithms,
[3]Université de Montréal, [4]CIFAR Fellow
rajsai24@gmail.com, pau.rodriguez@servicenow.com

Abstract

Transfer learning from large-scale pre-trained models has become essential for many computer vision tasks. Recent studies have shown that datasets like ImageNet are weakly labeled since images with multiple object classes present are assigned a single label. This ambiguity biases models towards a single prediction, which could result in the suppression of classes that tend to co-occur in the data. Inspired by language emergence literature, we propose multi-label iterated learning (MILe) to incorporate the inductive biases of multi-label learning from single labels using the framework of iterated learning. MILe is a simple yet effective procedure that builds a multi-label description of the image by propagating binary predictions through successive generations of teacher and student networks with a learning bottleneck. Experiments show that our approach exhibits systematic benefits on ImageNet accuracy as well as ReaL F1 score, which indicates that MILe deals better with label ambiguity than the standard training procedure, even when fine-tuning from self-supervised weights. We also show that MILe is effective reducing label noise, achieving state-of-the-art performance on real-world large-scale noisy data such as WebVision. Furthermore, MILe improves performance in class incremental settings such as IIRC and it is robust to distribution shifts. Code: https://github.com/rajeswar18/MILe

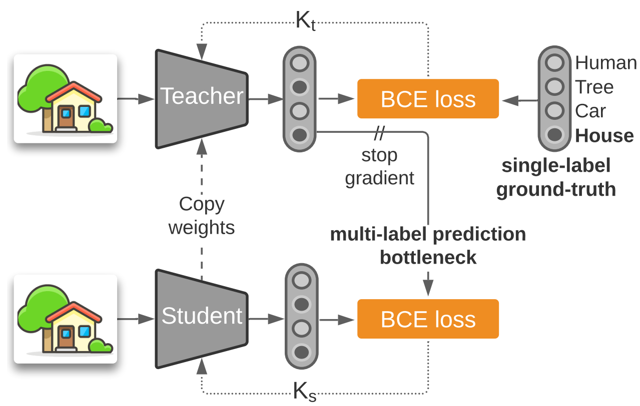


Figure 1. **Multi-label Iterated Learning (MILe)** builds a multi-label representation of the images from singly-labeled ground-truth. In this example, a model produces multi-label binary predictions for the next generation, obtaining *Car* and *House* for an image weakly labeled with *House*.

1. Introduction

Large-scale datasets with human-annotated labels have been central to the development of modern state-of-the-art neural network-based artificial perception systems [23, 24, 32]. Improved performance on ImageNet [17] has led to remarkable progress in tasks and domains that leverage ImageNet pretraining [11, 42, 70]. However, these weakly-annotated datasets and models tend to project a rich, multi-label reality into a paradigm that envisions one and only one label per image. This form of simplification often hinders model performance by asking models to predict a single label, when trained on real-world images that contain multiple objects.

Given the importance of the problem, there is growing recognition of single-labeled datasets as a form of weak supervision and an increasing interest in evaluating the limits of these singly-labeled benchmarks. A series of recent studies [8, 54, 56, 59, 65] highlight the problem of label ambiguity in ImageNet. In order to obtain a better estimate of model performance, Beyer et al. [8] and Shankar et al. [54] introduced multi-label evaluation sets. They identified softmax cross-entropy training as one of the main reasons for low multi-label performance since it promotes label exclusiveness. They also showed that replacing the softmax with sigmoid activations and casting the output as a set of binary classifiers results in better multi-label validation performance. Several other studies have explored ways to overcome the shortcomings in existing validation procedures by improving the pipelines for gathering labels [6, 48, 58].

In order to obtain a more complete description of images from weakly-supervised or semi-supervised data, a number

*Equal contribution.

of methods leverage a noisy signal such as pseudo-labels [65] or textual descriptions crawled from the web [47]. In this work, we observe that the process of building a rich representation of data from a noisy source shares some properties with the process of language emergence studied in the cognitive science literature. In particular, Kirby [29] proposed that structured language emerged from an inter-generational *iterated learning* process [29, 30, 31]. According to the theory, a compositional syntax emerges when agents learn by imitation from previous generations in the presence of a learning bottleneck. This bottleneck forces noisy fragments of the language to be forgotten when transmitted to new generations. Conversely, those fragments that can be reused and composed to enrich the language tend to be passed to subsequent generations. We show that the same procedure can be applied to settings that leverage a weak or noisy supervisory signal such as [47, 65] to build a richer description of images while reducing the noise.

In this work, we propose multi-label iterated learning (MILe) to learn to predict rich multi-label representations from weakly supervised (single-labeled) training data. We do so by introducing two different learning bottlenecks. First, we replace the standard convolutional neural network output softmax with a hard multi-label binary prediction. Second, we transmit these binary predictions through successive model generations, with a limited training iterations between each generation.

In our experiments, we demonstrate that MILe alleviates the label ambiguity problem by improving the F1 score of supervised and self-supervised models on the ImageNet ReaL [8] multi-label validation set. In addition, experiments on WebVision [37] show that iterated learning increases robustness to label noise and spurious correlations. Finally, we show that our approach can help in continual learning scenarios such as IIRC [1] where newly introduced labels co-occur with known labels. Our contributions are:

- We propose MILe, a multi-label iterated learning algorithm for image classification that builds a rich multi-label representation of data from weak single labels.
- We show that models trained with MILe are more robust to noise and perform better on ImageNet, ImageNet-ReaL, WebVision, and multiple setups such as supervised learning (Section 4.1), self-supervised fine-tuning and semi-supervised learning (Section 4.2), continual learning (Supplementary 2), and domain generalization (Supplementary 5).
- We provide insights on the predictions made by models trained with iterated learning (Section 4.3).

2. Related Work

It is known that weakly-labeled datasets such as ImageNet contain label ambiguity [6, 8, 54, 56, 59, 65] and label noise [49, 61]. Label ambiguity refers to the cases where only one of the multiple possible labels was assigned to the image. In order to evaluate how label ambiguity affects ImageNet classifiers, Beyer et al. [8] proposed ReaL, a curated version of the ImageNet validation set with multiple labels per image. They found that ImageNet classifiers tend to perform better on ReaL since it contains less label noise but they did not address the problem of inaccurate supervision during training where more than one correct class is present in the image. To deal with unfavorable training dynamics due to the mismatch between the multiplicity of object classes and the majority-aggregated single labels, Yun et al. [65] proposed to relabel the ImageNet training set. They obtained pixel-wise labels by finetuning an ensemble of large models pretrained on a large external dataset [57]. Although useful, undertaking such relabeling procedure for each dataset of interest is both laborious and unrealistic. In addition, it is not clear if the same relabeling approach could be used in larger, noisier databases such as WebVision [37], which contains 2.4M images downloaded from the internet and labels consisting of the queries used to download those images. In this work, we investigate the use of iterated learning on weak singly-labeled datasets as an alternative to relabeling in order to produce a multi-label output space. Different from existing methods, MILe uses neither external data nor additional relabeling procedures.

Knowledge Distillation Knowledge distillation is a technique commonly used in model compression [5, 9, 27]. In the vanilla setting, a large deep neural network is used as a teacher to train a smaller student network from its logits. In addition to model compression, knowledge distillation has been used to improve the generalization of student networks reusing distilled students as teachers [18] or distilling ensembles into a single model [2]. Gains have been observed even when the teacher and the student model are the same network, a regime commonly known as self-distillation [2, 46, 67]. Mobahi et al. [46] further showed that iterative self-distillation induces a strong regularization effect, with effects that are different from early stopping. Self-distillation has also been used to improve the generalization and robustness of semi-supervised models. Xie et al. [63] introduced noisy student for labeling unlabeled data during semi-supervised learning. While MILe also leverages teacher and student networks, it is fundamentally different from knowledge distillation approaches. The goal of knowledge distillation is to transmit all the knowledge of a teacher network to a student network. On the other hand, MILe trains a succession of short-lived teacher and student generations, thus creating an iterated learning bottleneck [29], to construct a new multi-label representation of the images from single labels. This goal is also different from the goal of noisy student, which is to label unlabeled data, and which is trained three times until convergence.

Iterated Learning The iterated learning hypothesis was first proposed by Kirby [29, 30] to explain language evolution via cultural transmission in humans. Languages need to be expressive and compressible to be effectively transmitted through generations. This learning bottleneck favors languages that are compositional as they can be easily and quickly learned by the offsprings and support generalization. Kirby et al. [31] conducted human experiments and mathematical modeling, which showed that iterated transmission of unstructured language results in convergence to a compositional language. Since then, it has seen many successful applications, especially in the emergent communication literature [15, 16, 22, 50]. In these settings, the learning bottleneck is induced by limiting the data or learning time of the student, which helps it to converge to a compositional language that is easier to learn [35]. The approach starts by training a *teacher network* with a small number of updates on the training set. A *student network* is then trained to imitate the teacher based on pseudo-multi-labels inferred from the input samples. The student then replaces the teacher and the cycle repeats with a frequency modulated by a learning budget. Iterated learning has also been used in the preservation of linguistic structure in addition to its emergence by Lu et al. [43, 44]. Furthermore, Vani et al. [62] successfully applied it for emergent systematicity in VQA. To the best of our knowledge, this is the first application of the iterated learning framework in the visual domain.

3. Method

We propose MILe to counter the problem of label ambiguity in singly-labeled datasets. We delineate the details of our approach to perform multi-label classification from weak singly-labeled ground truth.

Enforcing Multi-label Prediction. Singly-labeled datasets such as ImageNet usually represent their labels as one-hot vectors (all dimensions are zero except one). Training on these one-hot vectors forces models to predict a single class, even in the presence of other classes. Forcing models to predict a single class exposes them to biases in the image labeling process such as the preference for centered objects. Besides, constraining the model to output a single label per image limits the capability of perceptual models to capture all the content of the image accurately. In order to solve this problem, we propose to relax the model's output predictions from singly-labeled softmax prediction to multi-label binary prediction with sigmoids. Thus, we treat the singly-labeled classification problem as a set of independent binary classification problems. Since the ground-truth labels are still represented as one-hot vectors and training on them would still result in singly-labeled predictions, we propose an iterated learning procedure to bootstrap a multi-label pseudo ground truth.

Multi-label Iterated Learning. Our learning procedure is composed of two phases. In the first phase, a *teacher* model interacts with the single-labeled data to improve its predictions. The interaction is limited to a few iterations to prevent the binary classification model from overfitting to one-hot vectors. In the second phase, we leverage the acquired knowledge to train a different model, the *student*, on the multi-label predictions of the teacher. This yields a better initialization of the model for further iterations as we repeat this two-phased learning multiple times (see Alg. 1).

Specifically, we consider two parametric models, the teacher $f(.;\theta_\tau^T)$ and the student $f(.;\theta_\tau^S)$. Parameters of the teacher θ_τ^T are initialized using the student parameters θ_τ^S at iteration τ. First, we train the teacher for k_t learning steps on the labeled images from the dataset, obtaining $f(.;\theta_{\tau+1}^T)$. This constitutes the interaction phase of an iteration. We then move to the imitation phase, where we train the student to fit the teacher model for k_s steps, obtaining $f(.;\theta_{\tau+1}^S)$. This is done by training the student on the pseudo labels generated by the teacher on the data. Finally, we instantiate a new teacher by duplicating the parameters of this new student and iterate the process until convergence. In addition to yielding a smooth transition during the imitation phase, this procedure ensures that each iteration yields an improvement over the previous one (unless it is already optimal). Note that in the supervised learning regime we do not pseudo label any unlabeled data. In Sec. 4.2 we provide additional experiments showing that MILe can leverage unlabeled data in the semi-supervised learning regime.

Both the teacher and the student are trained on the same dataset $\mathcal{D}$ composed of input-label pairs $\{\mathcal{X},\mathcal{Y}\} \in \mathcal{D}$. We train the teacher to maximize the likelihood $p(\hat{y}=y|x,\theta) = \sigma(f(x,\theta))$, where $\hat{y}$ is the label predicted by the model, $y \in \mathcal{Y}$ is the true label, and σ is a normalization function such as the sigmoid. In order to alleviate the problem of label ambiguity, we consider $\mathcal{Y}$ a multi-label binary vector in $\mathbb{Z}_2^C$ where C is the number of classes and optimize the binary cross-entropy loss:

$$\mathcal{L}_{BCE} = -\frac{1}{B}\sum_{i=1}^{B}\frac{1}{C}\sum_{j=1}^{C} y_{i,j}\cdot log(\hat{y}_{i,j}) + (1-y_{i,j})\cdot log(1-\hat{y}_{i,j}), \tag{1}$$

where B is the number of samples in a batch when using batched stochastic gradient descent. We show in our experiments that iterated learning along with multi-label objective provides a strong inductive bias for modeling the effects of label ambiguity. Note that optimizing the binary cross-entropy on one-hot labels would not solve the label ambiguity problem. Thus, during each cycle, we train the teacher for a few iterations in order to prevent it from overfitting the one-hot ground truth. During student training, we threshold the teacher's output sigmoid activations to obtain multi-label pseudo ground-truth vectors $\tilde{y} = f(x, \theta^T) > \rho$. The threshold ρ is 0.25 unless otherwise stated.

The MILe Learning Bottleneck. Enforcing the imitation phase with some form of a learning budget is an essential component of the iterated learning framework [29]. This bottleneck regularizes the student model not to be amenable to the specific irregularities in the data. Kirby [29] argue that such a bottleneck enforces innate constraints on language acquisition. We believe that incorporating such a mechanism into the prediction models could prevent them from overfitting label noise [39], improving the quality of pseudo labels. There are two common ways to impose a learning bottleneck. One way is to allow a newly initialized student to only obtain the knowledge from a limited number of data instances generated by the teacher [29, 40]. Another is by limiting the number of student learning updates while imitating the teacher [43]. In our setting, we find it helpful to enforce the bottleneck via the number of learning updates.

As illustrated in Fig. 1 and Alg. 1, we iteratively refine a teacher network that is trained with the original labels and a student network that is trained with labels produced by the teacher. In order to prevent the student from overfitting the teacher, we restrict the amount of training updates [43] for each of the modules. Formally, let N be the size of the dataset, k_t be the number of training iterations of the teacher, and k_s the number of student iterations. In general, we set $k_t << N$ to prevent the teacher from overfitting one-hot labels and $k_s <= k_t$ to prevent the student from overfitting the teacher. In other words, each of our iterations is composed of two finite loops of (a) model improvement (teacher learning) and (b) model imitation (student learning).

Computational Cost. We train MILe for the same total number of epochs as standard supervised classification models. Thus, the total number of *backward* passes through the model (counting both the teacher and the student) is the same as the standard supervised training. Thus, the only additional computational cost comes from producing pseudo-labels with the teacher model. Moreover, the pseudo-labeling only happens once per teacher-student cycle and the network is in inference mode. Assuming $k_s + k_t = 10K$ (see Figure 3) and a batch size of 256, this inference pass only happens every 2.1 epochs for the ImageNet. Thus, the computational impact of MILe only constitutes a small fraction of the overall computational cost of training a neural network on the ImageNet. This computational cost could be easily compensated by skipping validation on alternate epochs or by validating in a different parallel process.

Algorithm 1 MILe

Require: **Initialize** Student network $\boldsymbol{\theta}_\tau^S$, $\tau = 0$. {*Prepare Iterated Learning*}
repeat
  Copy $\boldsymbol{\theta}_\tau^S$ to $\boldsymbol{\theta}_{\tau+1}^T$ {*Initialize Teacher*}
  for $i = 1$ **to** k_t **do**
    Sample a batch $(\boldsymbol{x_i}, \boldsymbol{y_i}) \in \mathcal{D}_{train}$
    $\hat{\boldsymbol{y}}_{\boldsymbol{i}} = f_{\boldsymbol{\theta}^T}(\boldsymbol{x_i})$
    $\boldsymbol{\theta}_{\tau+1}^T \leftarrow \boldsymbol{\theta}_{\tau+1}^T + \alpha \nabla \mathcal{L}^{BCE}(\boldsymbol{\theta}_{\tau+1}^T; \boldsymbol{y_i}, \hat{\boldsymbol{y}}_{\boldsymbol{i}})$ {*Update $\boldsymbol{\theta}^T$ to minimize L*}
  end for {*Finish Interactive Learning*}
  for $i = 1$ **to** k_s **do**
    Sample a batch $(\boldsymbol{x_i}, \boldsymbol{y_i}) \in \mathcal{D}_{train}$
    $\hat{\boldsymbol{y}}_{\boldsymbol{i}} = \sigma(f_{\boldsymbol{\theta}_{\tau+1}^T}(\boldsymbol{x_i})) > \rho$ {*Generate Pseudo Labels*}
    $\tilde{\boldsymbol{y}}_{\boldsymbol{i}} = f_{\boldsymbol{\theta}^S}(\boldsymbol{x_i})$
    $\boldsymbol{\theta}_\tau^S \leftarrow \boldsymbol{\theta}_\tau^S + \alpha \nabla \mathcal{L}^{BCE}(\boldsymbol{\theta}_\tau^S; \tilde{\boldsymbol{y}}_{\boldsymbol{i}}, \hat{\boldsymbol{y}}_{\boldsymbol{i}})$ {*Update $\boldsymbol{\theta}^S$ to minimize L*}
  end for {*Finish Imitation*}
  Copy $\boldsymbol{\theta}_\tau^S$ to $\boldsymbol{\theta}_{\tau+1}^S$
  $\tau \leftarrow \tau + 1$
until Convergence or maximum τ reached

4. Experiments

We provide experiments showing the effects of iterated learning in multiple setups. In Sec. 4.1, we study the robustness to label ambiguity and noise on ImageNet Real and WebVision. In Sec. **??**, we explore the benefits of iterated learning for domain generalization. In Sec. 4.2, we study the effect of MILe on models pre-trained with self-supervised objectives. Finally, in Sec. 4.3, we provide ablation experiments on the different hyperparameters as well as a more challenging synthetic setup with greater label ambiguity. Additional experiments in the Supplementary Material include a comparison with noisy student, multi-label learning on CelebA, and continual learning on IIRC.

4.1. Label Ambiguity and Noise

Datasets: We train our models on the standard ImageNet image classification benchmark [51], which is known to contain ambiguous labels [8]. Therefore, in addition to the validation set performance, we also report the performance on ReaL [8], an additional set of multi-labels for the ImageNet validation set gathered using a crowd-sourcing platform. ReaL contains a total of 57,553 labels for 46,837 images. We report results when using fractions of the total amount of training examples (i.e., 1%, 5%, 10%, 100%). To test the robustness of our method to label noise, we provide results on WebVision [37], which contains more than 2.4 million images crawled from the Flickr website and Google Images search. The same 1,000 concepts as the ImageNet ILSVRC 2012 dataset are used for querying images. It is worth noting that many ImageNet (ReaL) samples contain a single object and a single label. In Sec. 4.3, we explore the limits of MILe

Figure 2. **Qualitative results.** ReaL: original labels. Sigmoid: ResNet-50 with sigmoid output activations. MILe: multi-label iterated learning (ours).

on a synthetic dataset. In addition, we provide results on CelebA [41] in the supplementary material.

Baselines: We train a ResNet-18 and a ResNet-50 [23] model. Note that we favored vanilla ResNets over more advanced architectures and training procedures in order to focus on the advantages of MILe, rather than showing state-of-the-art results. We compare three different methods. (i) *Softmax*: standard softmax cross-entropy loss used to train the original ResNet backbone [23]. (ii) *Sigmoid*: we substitute the cross-entropy loss for a binary cross-entropy (BCE) loss. (iii) *MILe*: the proposed method as described in Sec. 3. For WebVision experiments, we also train an additional ResNet-50-D [26] backbone following more recent methodologies [64].

Metrics: We report accuracy on the original [51] and the ReaL [8] ImageNet validation set. ReaL is a multi-label dataset, so we calculate the accuracy as described by Beyer et al. [8]. Namely, we consider a top-1 prediction correct if it coincides with any of the ground-truth labels, i.e. ReaL-Acc $= \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i \cap Y_i| > 0$, where $\hat{y}_i$ is the predicted label for the ith sample, Y_i is the set of ReaL labels, and $|.|$ counts the the number of elements in a set. Additionally, we report the F1-score, which represents the proportion of correct predicted labels to the total number of actual and predicted labels, averaged across all examples: ReaL-F1 $= \frac{1}{N}\sum_{i=1}^{N}\frac{2\cdot TP_i}{2\cdot TP_i+FP_i+FN_i}$, where TP is the number of true positives, FP is the number of false positives, and FN is the number of false negatives. Finally, we report the label coverage, which indicates the total fraction of labels per sample predicted by the multi-label classifier. A number 1.15 indicates an additional 15% of labels was predicted.

ImageNet results. We report the results in Table 1. MILe surpasses baseline methods on all metrics and all fractions of training data. With Sigmoid, we observe a substantial improvement on ReaL-Acc of $\sim 2\%$ and $\sim 4\%$ for ResNet-18 and ResNet-50 respectively. This is in agreement with the results reported by Beyer et al. [8]. Incorporating iterative learning results in an extra $\sim 1\%$ performance improvement when using all the training data and up to 5% of ReaL-F1 when using a smaller fraction of the data. Interestingly, we find that using smaller fractions of data reduces the label coverage. We hypothesize that using a smaller fraction of the data leads to memorization and overfitting for the Softmax method and Sigmoid, which results in more confident predictions on a single class. Additional results focused on ReaL label recovery can be found in the supplementary material.

We report qualitative results in Fig. 2. As it can be seen, MILe produces more complete descriptions of the image, sometimes capturing labels that were not included in the ReaL ground truth. For instance, our method was able to detect a pickelhaube (pointy hat) that was not labeled in the ground truth.

WebVision results. We report results in Table 2 and put them in context with other state of the art. We adopt the same class rebalancing strategy as [36]. For all setups, we observe that MILe attains the best performance, up to 2 points better than methods using better architectures such as Inception-V3 [53]. We also validate the WebVision-trained model on the ImageNet validation set, outperforming the previous state of the art and keeping results consistent with the WebVision validation set. These results suggest that the iterated learning bottleneck acts as a regularizer that prevents the model from learning noisy labels which are more difficult to fit. This hypothesis is in agreement with Arpit et al. [4], Liu et al. [39], Zhang et al. [66], who showed that noise memorization happens later in the training procedure.

	ImageNet fraction:	1%	5%	10%	100%	1%	5%	10%	100%
Metric	Method	ResNet-50				ResNet-18			
Accuracy	Softmax	6.32	36.71	53.50	76.33	6.61	31.5	48.82	70.41
	ELR [39]	7.91	38.88	56.15	76.75	6.93	32.95	49.95	70.83
	Sigmoid	6.70	36.9	55.01	76.35	6.88	31.10	49.14	70.46
	MILe (ours)	**9.10**	**42.52**	**57.29**	**77.12**	**8.20**	**36.20**	**51.31**	**71.12**
ReaL-Acc	Softmax	7.19	42.55	60.21	82.76	8.80	35.88	55.11	77.77
	ELR [39]	8.78	44.24	63.13	83.07	8.92	38.08	56.13	78.85
	Sigmoid	8.38	46.04	62.96	83.22	9.04	37.66	57.52	81.01
	MILe (ours)	**11.50**	**48.36**	**65.42**	**83.75**	**9.18**	**41.65**	**58.57**	**81.52**
ReaL-F1	Softmax	6.77	40.51	57.33	78.5	8.28	34.20	52.51	73.83
	ELR [39]	7.83	42.45	58.52	78.5	8.41	35.52	53.22	73.41
	Sigmoid	7.17	41.11	58.46	78.61	8.39	33.56	52.12	73.85
	MILe (ours)	**10.76**	**45.02**	**62.11**	**79.89**	**8.55**	**38.49**	**53.80**	**74.48**
Label Coverage	Softmax	1.00	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	ELR [39]	1.00	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	Sigmoid	1.09	1.11	1.10	1.11	1.07	1.10	1.15	1.15
	MILe (ours)	1.05	1.08	1.09	1.16	1.06	1.07	1.12	1.17

Table 1. **ImageNet results.** The first row displays the fraction of the ImageNet data used to train the models. Softmax: Vanilla ResNet with softmax loss. Sigmoid: Vanilla ResNet trained for multi-label binary classification with single labels. ELR: early-learning regularization [39]. MILe: multi-label iterated learning. Label coverage refers to the fraction of additional labels predicted by each model. All the models are trained for 100 epochs.

Method	Architecture	WebVision Top-1	WebVision Top-5	ImageNet Top-1	ImageNet Top-5
CrossEntropy [60]	ResNet-50	66.4	83.4	57.7	78.4
MentorNet [28]	InceptionRes-V2	70.8	88.0	62.5	83.0
CurriculumNet [21]	Inception-V2	72.1	89.1	64.8	84.9
CleanNet [34]	ResNet-50	70.3	87.8	63.4	84.6
CurriculumNet [21, 60]	ResNet-50	70.7	88.6	62.7	83.4
SOM [60]	ResNet-50	72.2	89.5	65.0	85.1
Distill [69]	ResNet-50	-	-	65.8	**85.8**
MoPro (dec.) [36]	ResNet-50	72.4	89.0	65.7	85.1
Multimodal [53]	Inception-V3	73.15	89.73	-	-
Sigmoid	ResNet-50	72.1	89.5	65.4	85.0
MILe (ours)	ResNet-50	**75.2**	**90.3**	**67.1**	85.6
Initial Vanilla Model	ResNet-50-D	75.08	89.22	67.23	84.09
SCC [64]	ResNet-50-D	75.36	89.38	67.93	84.77
SCC+GBA [64]	ResNet-50-D	75.69	89.42	68.35	85.24
MILe (ours)	ResNet-50-D	**76.5**	**90.9**	**68.7**	**86.4**

Table 2. **WebVision results.** Methods are trained on Webvision-1000 and validated both on WebVision and ImageNet. MoPro (decoupled) is pre-trained on the same set as our method. CleanNet [34] and Distill [69] require data with clean annotations. dec: refers to "decoupled".

4.2. Self-supervised Fine-tuning

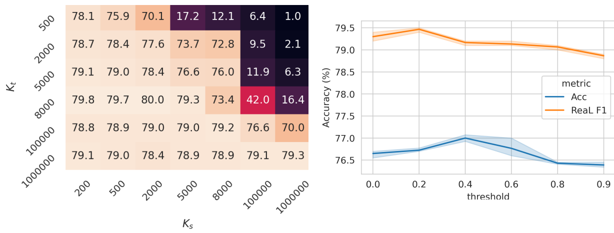
ImageNet's label ambiguity [8, 54, 56, 59, 65] might be problematic for fully-supervised methods but it is pos-

Method	ImageNet Validation 1%	ImageNet Validation 10%	ImageNet Validation 100%	ImageNet ReaL-F1 1%	ImageNet ReaL-F1 10%	ImageNet ReaL-F1 100%
SimCLR [12]	48.3	65.6	76.25	51.54	69.16	76.91
BYOL [20]	53.2	68.8	77.2	54.32	70.81	78.85
SwAV [10]	53.9	70.2	77.74	55.79	71.22	79.18
MoCo-v2 [14]	51.72	66.5	77.12	53.34	70.75	79.04
MILe (Ours) + [14]	**52.62**	**67.4**	**77.38**	**56.08**	**71.48**	**80.03**
SimCLR-v2-sk0 [13]	58.18	68.9	76.3	57.25	70.11	78.83
MILe (Ours) + [13] (sk0)	**61.85**	**70.5**	**77.29**	**60.49**	**72.76**	**79.38**
SimCLR-v2-sk1 [13]	64.7	72.4	78.7	62.77	74.21	79.43
MILe (Ours) + [13] (sk1)	**69.4**	**74.7**	**79.5**	**65.04**	**76.40**	**81.53**

Table 3. **Self-supervised finetuning.** The second row displays the fraction of ImageNet training data used for fine-tuning. Accuracy of top-1 predictions are used for reporting the numbers.

Method	Teacher	Label fraction 1%	Label fraction 10%
Distilled [13]	R50 (2×+SK)	69.0	75.1
Self-distilled [13]	R50 (1x+SK)	70.15	74.43
MILe (ours)	R50 (1x+SK)	**73.08**	**75.3**

Table 4. **Self- semi-supervised learning.** ImageNet top-1 accuracy for ResNet-50 (R50) distilled from a SimCLR [12] model. 2×: teacher has 2× parameters than the student.



(a) Iterations

(b) Threshold

Figure 3. **Ablation study.** Comparison between different iteration schedules. (a) Sweep over length of teacher training k_t and length of student training k_s. We report the ReaL-F1 score. (b) ReaL F-1 and accuracy scores for a threshold value sweep (ρ).

sible that self-supervised pre-training procedures such as MoCo [25] or SimCLR [12] are immune to it. We explore whether iterated learning improves the performance of self-supervised models in the fully- and semi-supervised fine-tuning regimes. We perform experiments on the ImageNet dataset and report validation accuracy and ReaL-F1 as described in Sec. 4.1.

Baselines. We report results with ResNet-50 pre-trained with SimCLR [12], SimCLR-v2 [13], BYOL [20], MoCo-v2 [14], and SwAV [19]. Results are reported after fine-tuning weights with 1%, 10%, and 100% of the ImageNet training set. We use the same data subsets as Chen et al. [13]. We incorporate the proposed iterative learning procedure in the fine-tuning process of MoCo-v2 and SimCLR-v2. For SimCLR-v2, we also tested the "sk1" variant which was improved with selective kernels [13, 38], while "sk0" is the vanilla version. For the semi-supervised learning experiments, we compare with SimCLR-v2's distillation experiments, where a teacher predicts pseudo-labels on unlabeled data. We compare with ResNet-50 (2$\times$+SK), where the teacher has 2$\times$ capacity than the student, and ResNet-50 (1$\times$+SK) where the teacher and the student are the same models.

Results. We report fine-tuning results in Table 3. Iterated learning improves the performance of MoCo-v2, SimCLR, and SimCLR-v2 for all fine-tuning data fractions. Interestingly, the improvement gap grows when using better self-supervised initializations. For example, the ReaL improvement from the best performing SimCLR-v2-sk1 with 100% of the validation data is 4.6% while it is around 3% for MoCo-v2 and SimCLR-v2-sk0. We hypothesize that more accurate models lead to better teachers, improving the overall performance of the iterated learning procedure.

We report semi-supervised learning results in Table 4. Iterated learning performs 2.9% better with 1% of the training labels and 0.9% with 10% of the training labels when compared with the self-distillation procedure presented in SimCLR-v2 [13]. Interestingly, we find that iterated learning attains better performance than distilling from a teacher twice the size of the student.

4.3. Analysis

In this section we explore the behavior of MILe under different hyperparameter settings as well as more challenging setups with synthetic data.

Number of Iterations. We investigate the effect of the number of teacher iterations (k_t) and student iterations (k_s) per cycle on the final performance (Fig. 3a). We report the ReaL-F1 for different k_t values (rows) and k_s values (columns). In general, we find that good performance can be achieved with a wide range of k_t and k_s combinations. The best performance is achieved with smaller values of k_t and k_s. Extreme values of k_t and k_s lead to lower performance, with the model being most sensitive to large values of k_s (dark regions). This is expected since a small k_t would let the imitation phase constantly disrupt supervised learning via interaction with the data, while a large k_t does not reap the benefits of distillation. For a given k_t we find that the optimal k_s lies in the mid-range and the other way around. Regarding the influence of the dataset size, we observe that it mostly influences the optimal number of teacher iterations (k_t). We hypothesize that it takes few iterations for the teacher to overfit small datasets, which leads to one-hot predictions and prevents the model from learning a multi-label hierarchy.

Pseudo-label Threshold Ablation Study In this section, we conduct an ablation study on the threshold value (ρ) used by MILe to produce multi-pseudo-labels from sigmoid output activations (see Section 3 and Algorithm 1). Fig. 3b shows the validation accuracies and ReaL-F1 scores for different threshold values. Lower thresholds bias the student towards producing multi-label outputs, even for low-confidence classes. Larger threshold values make the student tend towards singly-labeled prediction, only predicting labels for which the confidence is high. In the extreme, a high threshold constrains the teacher to predict empty label vectors. Interestingly, we find that lower threshold values result in higher ReaL-F1 score and better accuracy. In fact, the Real-F1 score benefits from lower ρ than the accuracy. This is due to the fact that lower thresholds increase the number of predicted labels per image, which improves the recall in multi-label evaluations.

Multi-label MNIST Many images in the real world datasets like WebVision or ImageNet contain a single object, which biases MILe towards predicting a small number of objects per image. In order to explore the limits of MILe, we begin by designing a controlled experiment on a synthetic dataset where most samples contain multiple classes. Each sample consists of a 3×3 grid of randomly sampled MNIST

	F1@0.25	F1@0.5
Softmax	28.69	28.69
Sigmoid	29.10	28.67
MILe (ours)	**41.35**	**34.32**

Table 5. **Results on multi-label MNIST.** The first column displays the F1 score when the threshold for positive labels is set to 0.25 and the second column shows the F1 score for a threshold of 0.5.

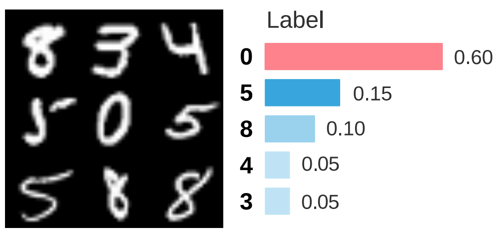


Figure 4. **Multi-MNIST.** The center digit has a probability of 0.6 to be chosen as the label for the whole grid.

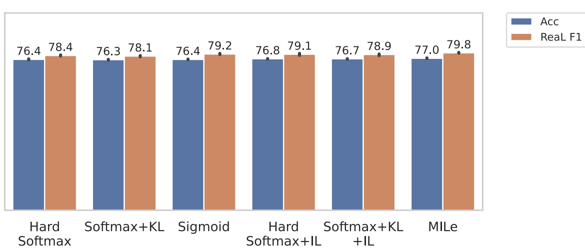


Figure 5. Comparison between different distillation schedules and MILe. We report the Accuracy and ReaL-F1 score.

digits [33]. For each grid, its single label corresponds to the center digit with probability 0.6 while the 8 remaining digits are sampled with probability 0.05 each (see Fig. 4). Note that, similar to the ImageNet, digits of the same class can repeat in the grid. However, the probability of having a 3×3 grid with the same digit repeated in each position is 10^{-9}.

Results are shown in Table 5. We observe that MILe attains up to 12% better F1 score than the Softmax and Sigmoid baselines. It is worth noting that the improvement is most significant when thresholding the sigmoid output predictions to 0.25. Interestingly, for this experiment, we found the best threshold to produce multi-pseudo-labels from the teacher output to be ($\rho = 0.1$). Having a low threshold biases the student towards producing multi-label outputs. We find these results encouraging and we believe that better performance could be attained by improving the pseudo-multi-label generation strategy. We plan to explore these new strategies in future work.

Contribution of Self-Distillation and Iterated Learning Here, we study of the effect of the multi-label distillation algorithm on the iterative procedure. We compare soft distillation (softmax + KL loss) with hard distillation (argmax + CE), and MILe (sigmoid + threshold + BCE) with and without iterated learning in Fig. 5. We compare the effect on two and many iterations. Hard labels outperform soft labels when training with many iterations. We provide an ablation of iterated learning with nosiy-student [63] distillation procedure depicted in Fig. 8 of the supplementary material.

5. Discussion

We introduce multi-label iterated learning (MILe) to address the problem of label ambiguity and label noise in popular classification datasets such as ImageNet. MILe leverages iterated learning to build a rich supervisory signal from weak supervision. It relaxes the singly-labeled classification problem to multi-label binary classification and alternates the training of a teacher and a student network to build a multi-label description of an image from single labels. The teacher and the student are trained for few iterations in order to prevent them from overfitting the singly-labeled noisy predictions. MILe improves the performance of image classifiers for the singly-labeled and multi-label problems, domain generalization, semi-supervised learning, and continual learning on IIRC. A possible limitation, which is inherent to iterated learning [43], is choosing the correct length of teacher (k_t) and student iterations (k_s). However, our ablation experiments suggest that the proposed procedure is beneficial for a wide range of k_t and k_s values (Sec. 4.3). MILe also depends on the threshold value ρ, which we use to produce pseudo-labels from the teacher's outputs. However, we found encouraging that low values of ρ improve the performance of the classifiers, indicating that predicting multiple labels is beneficial. With respect to the computational cost, we found that the impact of MILe is lower than the validation phase of the models (see Sec. 3). Overall, we found that iterated learning improves the performance of models trained with weakly labeled data, helping them to overcome problems related to label ambiguity and noise.

Broader impact and future work. Our approach is built on the hypothesis that the world is structured along objects and the fact that images result from the composition of those objects. We believe that our work could be applied to other tasks that build on the same assumptions such as object detection, segmentation, and multiple-instance learning. In these cases we hope approaches like MILe could open the door to leverage large amounts of webly supervised data to improve on these tasks.

References

[1] M. Abdelsalam, M. Faramarzi, S. Sodhani, and S. Chandar. Iirc: Incremental implicitly-refined classification. *CVPR*, 2021.

[2] Z. Allen-Zhu and Y. Li. Towards understanding ensemble, knowledge distillation and self-distillation in deep learning. *arXiv preprint arXiv:2012.09816*, 2020.

[3] M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[4] D. Arpit, S. Jastrzębski, N. Ballas, D. Krueger, E. Bengio, M. S. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio, et al. A closer look at memorization in deep networks. In *ICML*, 2017.

[5] L. J. Ba and R. Caruana. Do deep nets really need to be deep? *arXiv preprint arXiv:1312.6184*, 2013.

[6] A. Barbu, D. Mayo, J. Alverio, W. Luo, C. Wang, D. Gutfreund, J. B. Tenenbaum, and B. Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. In *NeurIPS*, 2019.

[7] E. Bekele and W. Lawson. The deeper, the better: Analysis of person attributes recognition. In *International Conference on Automatic Face Gesture Recognition*, 2019.

[8] L. Beyer, O. J. Hénaff, A. Kolesnikov, X. Zhai, and A. v. d. Oord. Are we done with imagenet? *arXiv preprint arXiv:2006.07159*, 2020.

[9] C. Buciluă, R. Caruana, and A. Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD*, pages 535–541, 2006.

[10] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020.

[11] J. Carreira and A. Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. *CVPR*, 2017.

[12] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pages 1597–1607, 2020.

[13] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint:2006.10029*, 2020.

[14] X. Chen, H. Fan, R. Girshick, and K. He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

[15] M. Cogswell, J. Lu, S. Lee, D. Parikh, and D. Batra. Emergence of compositional language with deep generational transmission. *arXiv preprint arXiv:1904.09067*, 2019.

[16] G. Dagan, D. Hupkes, and E. Bruni. Co-evolution of language and agents in referential games. *arXiv preprint arXiv:2001.03361*, 2020.

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[18] T. Furlanello, Z. Lipton, M. Tschannen, L. Itti, and A. Anandkumar. Born again neural networks. In *ICML*, 2018.

[19] P. Goyal, M. Caron, B. Lefaudeux, M. Xu, P. Wang, V. Pai, M. Singh, V. Liptchinsky, I. Misra, A. Joulin, et al. Self-supervised pretraining of visual features in the wild. *arXiv preprint arXiv:2103.01988*, 2021.

[20] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.

[21] S. Guo, W. Huang, H. Zhang, C. Zhuang, D. Dong, M. Scott, and D. Huang. Curriculumnet: Weakly supervised learning from large-scale web images. In *ECCV*, 2018.

[22] S. Guo, Y. Ren, S. Havrylov, S. Frank, I. Titov, and K. Smith. The emergence of compositional languages for numeric concepts through iterated learning in neural agents, 2019.

[23] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[24] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick. Mask r-cnn. *ICCV*, 2017.

[25] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019.

[26] T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, and M. Li. Bag of tricks for image classification with convolutional neural networks. In *CVPR*, pages 558–567, 2019.

[27] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[28] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, and L. Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *ICML*, 2018.

[29] S. Kirby. Spontaneous evolution of linguistic structure-an iterated learning model of the emergence of regularity and irregularity. *IEEE Transactions on Evolutionary Computation, 5(2):102–110*, 2001.

[30] S. Kirby. Natural language from artificial life. *Artificial life, 8(2): 185–215*, 2002.

[31] S. Kirby, T. Griffiths, and K. Smith. Iterated learning and the evolution of language. *Current opinion in neurobiology, 28:108–114*, 2014.

[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012.

[33] Y. LeCun and C. Cortes. MNIST handwritten digit database. 2010. URL http://yann.lecun.com/exdb/mnist/.

[34] K.-H. Lee, X. He, L. Zhang, and L. Yang. Cleannet: Transfer learning for scalable image classifier training with label noise. In *CVPR*, 2018.

[35] F. Li and M. Bowling. Ease-of-teaching and language structure from emergent communication. In *NeurIPS*, 2019.

[36] J. Li, C. Xiong, and S. C. Hoi. Mopro: Webly supervised learning with momentum prototypes. *ICLR*, 2021.

[37] W. Li, L. Wang, W. Li, E. Agustsson, and L. Gool. Webvision database: Visual learning and understanding from web data. *ArXiv*, 2017.

[38] X. Li, W. Wang, X. Hu, and J. Yang. Selective kernel networks. In *CVPR*, pages 510–519, 2019.

[39] S. Liu, J. Niles-Weed, N. Razavian, and C. Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *NeurIPS*, 2020.

[40] W. Liu, Z. Liu, H. Wang, L. Paull, B. Schölkopf, and A. Weller. Iterative teaching by label synthesis, 2021.

[41] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *ICCV*, 2015.

[42] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[43] Y. Lu, S. Singhal, F. Strub, A. Courville, and O. Pietquin. Countering language drift with seeded iterated learning. In *ICML*, 2020.

[44] Y. Lu, S. Singhal, F. Strub, O. Pietquin, and A. Courville. Supervised seeded iterated learning for interactive language learning. In *EMNLP*, 2020.

[45] M. Masana, X. Liu, B. Twardowski, M. Menta, A. D. Bagdanov, and J. van de Weijer. Class-incremental learning: survey and performance evaluation. *arXiv preprint arXiv:2010.15277*, 2020.

[46] H. Mobahi, M. Farajtabar, and P. L. Bartlett. Self-distillation amplifies regularization in hilbert space, 2020.

[47] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision. *CoRR*, 2021.

[48] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar. Do imagenet classifiers generalize to imagenet? *arXiv*, 2019.

[49] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar. Do imagenet classifiers generalize to imagenet? In *ICML*, 2019.

[50] Y. Ren, S. Guo, M. Labeau, S. B. Cohen, and S. Kirby. Compositional languages emerge in a neural iterated learning model. In *ICLR*, 2020.

[51] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, pages 211–252, 2015.

[52] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. 2015.

[53] M. Shah, K. Viswanathan, C.-T. Lu, A. Fuxman, Z. Li, A. Timofeev, C. Jia, and C. Sun. Inferring context from pixels for multimodal image classification. In *CIKM*. ACM, 2019. ISBN 9781450369763.

[54] V. Shankar, R. Roelofs, H. Mania, A. Fang, B. Recht, and L. Schmidt. Evaluating machine accuracy on imagenet. In *ICML*, 2020.

[55] J. Speth and E. M. Hand. Automated label noise identification for facial attribute recognition. In *CVPR Workshops*, pages 25–28, 2019.

[56] P. Stock and M. Cisse. Convnets and imagenet beyond accuracy: Understanding mistakes and uncovering biases. In *ECCV*, pages 498–512, 2018.

[57] C. Sun, A. Shrivastava, S. Singh, and A. Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *ICCV*, 2017.

[58] D. Tsipras, S. Santurkar, L. Engstrom, A. Ilyas, and A. Madry. From imagenet to image classification: Contextualizing progress on benchmarks. In *ICML*, 2020.

[59] D. Tsipras, S. Santurkar, L. Engstrom, A. Ilyas, and A. Madry. From imagenet to image classification: Contextualizing progress on benchmarks. In *ICML*, 2020.

[60] Y. Tu, L. Niu, D. Cheng, and L. Zhang. Protonet: Learning from web data with memory. *CVPR*, 2020.

[61] G. Van Horn, S. Branson, R. Farrell, S. Haber, J. Barry, P. Ipeirotis, P. Perona, and S. Belongie. Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection. In *CVPR*, 2015.

[62] A. Vani, M. Schwarzer, Y. Lu, E. Dhekane, and A. Courville. Iterated learning for emergent systematicity in vqa. In *ICLR*, 2021.

[63] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le. Self-training with noisy student improves imagenet classification. In *CVPR*, pages 10687–10698, 2020.

[64] J. Yang, L. Feng, W. Chen, X. Yan, H. Zheng, P. Luo, and W. Zhang. Webly supervised image classification with self-contained confidence. *ECCV*, 2020.

[65] S. Yun, S. J. Oh, B. Heo, D. Han, J. Choe, and S. Chun. Relabeling imagenet: from single to multi-labels, from global to localized labels. In *CVPR*, 2021.

[66] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

[67] L. Zhang, J. Song, A. Gao, J. Chen, C. Bao, and K. Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *ICCV*, pages 3713–3722, 2019.

[68] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. Bourdev. Panda: Pose aligned networks for deep attribute modeling. In *CVPR*, 2014.

[69] Z. Zhang, H. Zhang, S. O. Arik, H. Lee, and T. Pfister. Distilling effective supervision from severe label noise. In *CVPR*, 2020.

[70] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *CVPR*, 2017.