OVERCOMING LABEL AMBIGUITY WITH MULTI-LABEL ITERATED LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Transfer learning from ImageNet pre-trained models has become essential for many computer vision tasks. Recent studies have shown that ImageNet includes label ambiguity, where 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. Recent approaches have explored either fixing the evaluation datasets or using costly procedures to relabel the training data. In this work, 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 alternates training a teacher and a student network with binary predictions to build a multi-label description of the images. Experiments on ImageNet show that MILe achieves higher accuracy and ReaL score than when using the standard training procedure, even when fine-tuning from self-supervised weights. We also show that MILe is effective for real-world large-scale noisy data such as WebVision. Furthermore, MILe improves performance in class incremental settings such as IIRC and is robust to distribution shifts.

1 INTRODUCTION

Large-scale datasets with human-annotated labels have been central to the development of modern state-of-the-art artificial perception systems [34, 25, 26]. Improved performance on ImageNet [18] has led to further progress in tasks and domains that leverage ImageNet pretraining [12, 44, 72]. However, annotated datasets and models tend to project a multi-label reality onto single-label images. This often hinders the model's performance by asking models to predict a single label, especially when trained on real-world images that contain multiple objects of interest.

Given the importance of the problem, there is an increasing interest in evaluating the limits of single-label benchmarks. A series of recent studies [58, 61, 56, 9, 67] highlight the problem of label ambiguity in ImageNet. In order to obtain a better estimate of model performance, Beyer et al. [9], Shankar et al. [56] 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, 60, 50]. Although multi-way training objectives improve validation performance, models are still trained to predict a single label per image. Consequently, Yun et al. [67] introduced a multi-label training set for ImageNet. They leveraged pixel-wise predictions from an ensemble of large models pretrained on external data. Models trained using these dense labels show competitive performance on both standard and multi-label ImageNet evaluations [9]. More recently, Radford et al. [49] (CLIP) demonstrated that learning from natural language descriptions rather than single-label supervision leads to robust and transferable representations. However, the dataset used to train CLIP has not been released yet, and ImageNet-based single-label training has not declined in popularity.

While multi-label training alleviates the problem of label ambiguity, the relabeling effort was only made for ImageNet, and relabeling all existing datasets with multi-labels is a daunting task. In this work, we observe that models trained for multi-label binary classification with singly-labeled ground truth tend to produce multi-label predictions in ambiguous images during the first training

iterations (before overfitting). We propose to leverage these multi-label predictions as pseudo labels to enrich the supervision signal of student networks in an iterated self-distillation procedure. We aim to produce a multi-label representation of the images when only a single label is provided during training.

Iterated learning is an algorithmic procedure first proposed in the field of cognitive science to model the emergence and evolution of language structure [31, 33]. During iterated learning, a compositional syntax emerges when agents learn by imitation from previous generations in the presence of a learning bottleneck. We show that the same procedure can lead to the emergence of a multi-label description of images from single labels. 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.

In this work, we propose multi-label iterated learning (MILe) as a solution to build better predictors by addressing the problem of label ambiguity. In our experiments, we demonstrate that iterated learning improves the performance of supervised and self-supervised models on the ImageNet and ImageNet ReaL [9] validation sets. In addition, experiments on WebVision [40] 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 prevents models from making single-label predictions in the presence of multiple objects from multiple classes.
- We provide quantitative and qualitative insights on the predictions made by models trained with iterated learning.
- We show that models trained with MILe are more robust and achieve competitive performance on ImageNet, ImageNet-ReaL, WebVision, and multiple setups such as supervised learning, self-supervised learning, semi-supervised learning, and continual learning.

2 Related Work

Learning with label noise ImageNet has become the standard image classification benchmark [53], and its importance has motivated several studies assessing its reliability. For instance, it is known that ImageNet contains label ambiguity [58, 61, 56, 9, 67, 6] and label noise [63, 51]. 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. [9] 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. [67] 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 [59]. 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 [40], 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 single-label 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 [10, 29, 5]. 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 [19] 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 [48, 69, 2]. Self-distillation has also been used to improve the generalization and robustness of semi-supervised models. Xie et al. [65] proposed to cycle teacher and student training. Empirically, they found that the

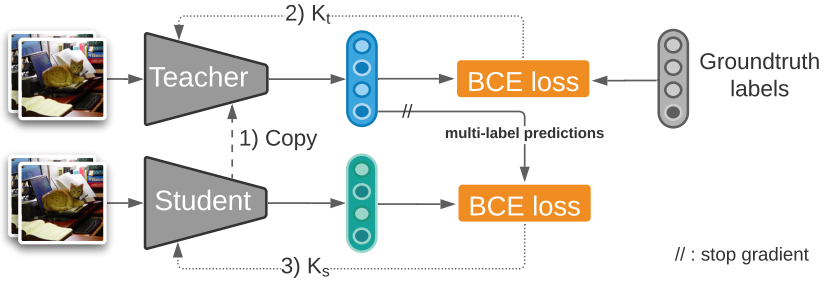


Figure 1: **MILe.** 1) Teacher and student are initialized with the same weights. 2) The teacher model is trained with ground truth labels for k_t iterations. 3) The student model is trained from the teacher pseudo-label predictions for k_s iterations. Finally, the teacher is initialized again with the student weights (1) and the process is repeated until convergence.

cycles of self-distillation brought diminishing returns after three iterations. In this work, we propose following the iterated learning paradigm by performing many cycles over the course of training and using a limited number of iterations per cycle to create a learning bottleneck. It is worth noting that Xie et al. [65] trained their model three times until convergence, which requires significantly more computation than iterated learning, which is only trained once until convergence.

Iterated Learning The iterated learning hypothesis was first proposed by Kirby [31; 32] 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. [33] 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 [24, 52, 16, 17]. 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 [38]. Iterated learning has also been used in the preservation of linguistic structure in addition to its emergence by Lu et al. [45; 46]. Furthermore, Vani et al. [64] 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 single-label datasets. We delineate the details of our approach to perform multi-label classification from single-label ground truth.

Enforcing multi-label prediction. ImageNet is a single-label dataset and thus, labels are usually represented 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 single-label softmax prediction to multi-label binary prediction with sigmoids. Thus, we treat the single-label 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 single-label 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. 4.4 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:

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

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 0.5 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 [31]. This bottleneck regularizes the student model not to be amenable to the specific irregularities in the data. Kirby [31] 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 [42], improving the quality of pseudo labels. Predominantly, there have been two 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 [31]. Another is by limiting the number of learning updates that the student can perform while imitating the teacher [45]. 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 [45] 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).

4 Experiments

We provide experiments showing the effects of iterated learning in multiple setups. We focus our study on ImageNet and WebVision [40]. In Sec. 4.1, we study the robustness to noise on ImageNet Real and WebVision. In Sec. 4.2, we test the proposed method on IIRC, a continual learning benchmark with growing multi-label space. In Sec. 4.3, we explore the benefits of iterated learning for domain generalization. In Sec. 4.4, we study the effect of MILe on models pre-trained with self-supervised objectives. Finally, in Sec. 4.5, we provide ablation experiments and comparisons with other iterative learning approaches such as self-distillation and noisy student.

Algorithm 1 MILe

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

4.1 Label Ambiguity and Noise

Datasets: We train our models on the standard ImageNet image classification benchmark [53], which is known to contain ambiguous labels [9]. Therefore, in addition to the validation set performance, we also report the performance on ReaL [9], 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 [40], 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 App. A.1, we explore the limits of MILe on a synthetic dataset. In addition, we provide results on CelebA [43] in App. A.6.

Baselines: We train a ResNet-18 and a ResNet-50 [25] using three different methods. (i) *Softmax*: standard softmax cross-entropy loss used to train the original ResNet backbone [25]. (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 [28] backbone following recent methodologies [66].

Metrics: We report accuracy on the original [53] and the ReaL [9] ImageNet validation set. ReaL is a multi-label dataset, so we calculate the accuracy as described by Beyer et al. [9]. 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. [9]. 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 App. A.2.

	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
	Sigmoid	6.70	36.9	55.01	76.35	6.88	31.1	49.14	70.46
	MILe (ours)	**9.10**	**42.52**	**57.29**	**77.12**	**8.2**	**36.2**	**51.31**	**71.12**
ReaL-Acc	Softmax	7.19	42.55	60.21	82.76	8.80	35.88	55.11	77.77
	Sigmoid	8.38	46.04	62.96	83.22	9.04	37.66	57.52	81.01
	MILe (ours)	**11.5**	**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
	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**	**53.8**	**74.48**
Label Coverage	Softmax	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: ResNet trained for multi-label binary classification with single labels. 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.

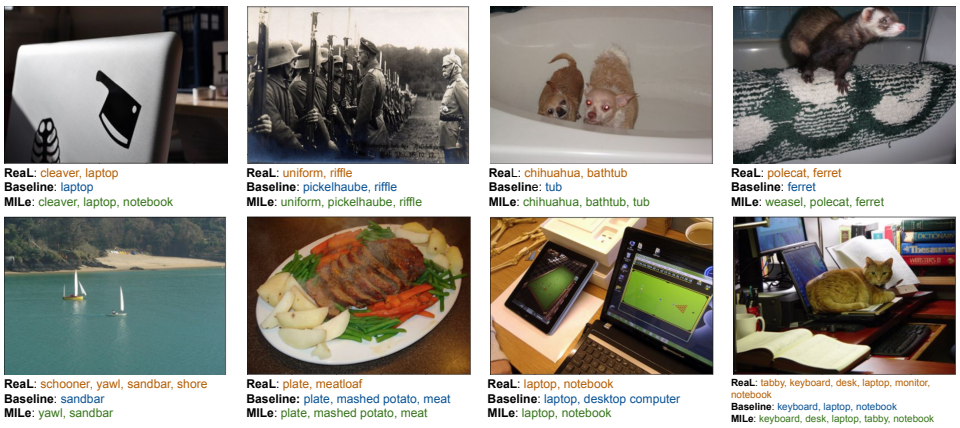


Figure 2: **Qualitative results.** ReaL: original labels. Sigmoid: ResNet-50 with sigmoid output activations. MILe: multi-label iterated learning (ours).

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.

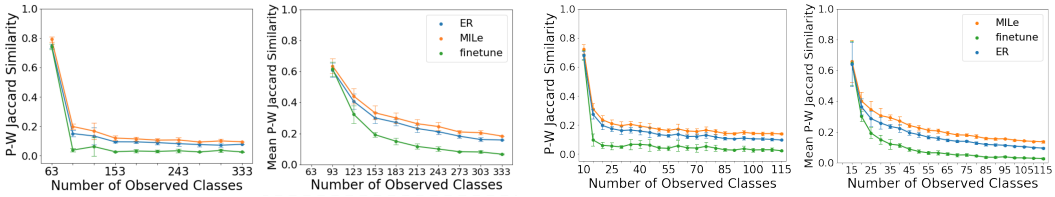
WebVision results. We report results in Table 2 and put them in context with other state of the art. For all setups, we observe that MILe attains the best performance, up to 2 points better than methods using better architectures such as Inception-V3 [55]. We also validate the WebVision-trained model 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], Zhang et al. [68], Liu et al. [42], who showed that noise memorization happens later in the training procedure.

4.2 IIRC BENCHMARK

We explore whether MILe can incrementally learn an increasingly complex class hierarchy by teaching previously seen tasks to new generations. We experiment with Incremental Implicitly-Refined Classification (IIRC) [1], an extension to the class incremental learning setup [47] where the incoming batches of classes have two granularity levels, e.g. a coarse and a fine label. Labels are

Method	Architecture	WebVision		ImageNet	
		Top-1	Top-5	Top-1	Top-5
CrossEntropy [62]	ResNet-50	66.4	83.4	57.7	78.4
MentorNet [30]	InceptionResNet-V2	70.8	88.0	62.5	83.0
CurriculumNet [23]	Inception-V2	72.1	89.1	64.8	84.9
CleanNet [37]	ResNet-50	70.3	87.8	63.4	84.6
CurriculumNet [23, 62]	ResNet-50	70.7	88.6	62.7	83.4
SOM [62]	ResNet-50	72.2	89.5	65.0	85.1
Distill [71]	ResNet-50	-	-	65.8	**85.8**
MoPro (decoupled) [39]	ResNet-50	72.4	89.0	65.7	85.1
Multimodal [55]	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 [66]	ResNet-50-D	75.36	89.38	67.93	84.77
SCC+GBA [66]	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 [37] and Distill [71] require data with clean annotations.



(a) IIRC-ImageNet (b) IIRC-CIFAR10

Figure 3: **IIRC evaluation**. (a) Average performance on IIRC-ImageNet-lite. (b) Average performance on IIRC-CIFAR10. We run experiments on five different task configurations and report the mean and standard deviation. Left: average performance when the tasks are equally weighted irrespective of how many samples exist per task. Right: average performance over the number of samples. In this case, the first task has more weight since it is larger in the number of samples.

seen one at a time, and fine labels for a given coarse class are introduced after that coarser class is visited. The goal is to incorporate new finer-grained information into existing knowledge in a similar way as humans learn different breeds of dogs after learning the concept of dog.

Results. Following the procedure described by Abdelsalam et al. [1], we train a ResNet-50 on ImageNet and a reduced ResNet-32 on CIFAR100. Also following Abdelsalam et al. [1], we compare with an *experience replay* (ER) baseline and a *finetune* lower-bound. We report the model's overall performance after training until task i as the precision-weighted Jaccard similarity between the model predictions and the ground-truth multi-labels over all classes encountered so far. See App. A.3 for details on the metric. We report IIRC-ImageNet-lite evaluation scores in Fig. 3a and CIFAR in Fig. 3b. In all cases, we find that iterative learning increases the performance with respect to the ER baseline by a constant factor. This suggests that MILe helps prevent forgetting previously seen labels by propagating them through the iterated learning procedure.

4.3 Domain Generalization

A common problem of machine-learning models is that they tend to fail when presented with out-of-distribution data [7]. Arjovsky et al. [3] claimed that this happens due to models relying on spurious correlations rather than the causal factors of the data. Thus, we investigate whether iterative learning can reduce the effect of spurious correlations by allowing the model to produce independent predictions of the two correlated factors. Following Arjovsky et al. [3], we perform experiments on ColoredMNIST [3], a version of MNIST where the color of the digits is spuriously correlated with their value. The spurious correlation is removed at test time, i.e. colors are assigned randomly, to

Method	ImageNet Validation			ImageNet ReaL-F1		
	1%	10%	100%	1%	10%	100%
SimCLR [13]	48.3	65.6	76.25	51.54	69.16	76.91
BYOL [21]	53.2	68.8	77.2	54.32	70.81	78.85
SwAV [11]	53.9	70.2	77.74	55.79	71.22	79.18
MoCo-v2 [15]	51.72	66.5	77.12	53.34	70.75	79.04
MILe (Ours) + MoCo-v2	**52.62**	**67.4**	**77.38**	**56.08**	**71.48**	**80.03**
SimCLR-v2-sk0 [14]	58.18	68.9	76.3	57.25	70.11	78.83
MILe (Ours) + SimCLR-v2-sk0	**61.85**	**70.5**	**77.29**	**60.49**	**72.76**	**79.38**
SimCLR-v2-sk1 [14]	64.7	72.4	78.7	62.77	74.21	79.43
MILe (Ours) + SimCLR-v2-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 top-1 predictions are used for reporting the numbers.

Method	CMNIST	CMNIST+
ERM	51.6± 0.1	51.1 ± 0.1
IRM [3]	51.8± 0.1	51.2 ± 0.2
REx [35]	51.6± 0.1	51.2 ± 0.2
MILe (ours)	51.8± 0.1	**53.5** ± 0.6

Table 4: **OOD generalization** on ColoredMNIST [3] (CMNIST), which consists in predicting digits and ColoredMNIST+, which consists in color or digit prediction.

Method	Teacher	Label fraction	
		1%	10%
Distilled [14]	R50 (2×+SK)	69.0	75.1
Self-distlled [14]	R50 (1x+SK)	70.15	74.43
MILe (ours)	R50 (1x+SK)	**73.08**	**75.3**

Table 5: **Self- semi-supervised learning.** ImageNet top-1 accuracy for ResNet-50 (R50) distilled from a SimCLR [13] model. 2×: teacher has 2× parameters than the student.

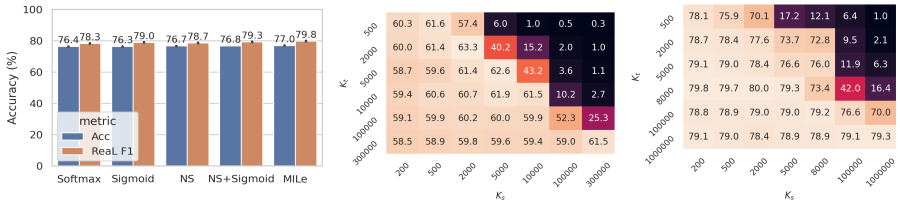
reveal whether models are affected by color. During training, we add an extra color classification task consisting of solid color images. For each task, models are either asked to predict the color or the digit but never both. This setup brings ColoredMNIST closer to ImageNet's label ambiguity problem, where labels are biased towards foreground (cow on a beach) but backgrounds (beaches) are also part of the classification problem. We call this setup ColoredMNIST+ (details in App. A.4).

Results. We compare with invariant risk minimization (IRM) [3] and risk extrapolation (REx) [35] based on the DomainBed implementation [22]. These two approaches leverage differences between multiple environments, with different levels of correlation between digit and color, to become invariant to spurious attributes.We report results in Table 4. MILe surpasses REx by 2 points. Interestingly, even though ERM and IRM are also required to solve the color classification task, only iterated learning is able to use it to improve performance. Although the color and digit prediction tasks are mutually exclusive, during iterated learning the teacher produces labels for both tasks simultaneously and thus the student learns to predict the color even for images that contain a digit. This helps the model to learn that these are two independent attributes, boosting its performance.

4.4 Self-supervised Fine-tuning

ImageNet's label ambiguity [58, 61, 56, 9, 67] might be problematic for fully-supervised methods but it is possible that self-supervised pre-training procedures such as MoCo [27] or SimCLR [13] 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 [13], SimCLR-v2 [14], BYOL [21], MoCo-v2 [15], and SwAV [20]. Results are reported after fine-tuning weights with 1%, 10%, and 100% of the ImageNet training set. 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 [41, 14], 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×+SK), where the teacher has 2× capacity than the student, and ResNet-50 (1×+SK) where the teacher and the student are the same models.



(a) Noisy student (NS) (b) 10% data (c) 100% data

Figure 4: **Ablation study.** Comparison between different iteration schedules. (a) Comparison with noisy student (NS). (b)(c) Sweep over length of interactive learning phase k_t and length of imitation phase k_s. We report the ReaL-F1 score for 10% (b) and 100% (c) data fraction.

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 iterated learning teachers, improving the overall performance of the iterated learning procedure.

We report semi-supervised learning results in Table 5. 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 [14]. Interestingly, we find that iterated learning attains better performance than distilling from a teacher twice the size of the student.

4.5 Ablation Study

We investigate the sensitivity to the number of teacher and student updates per cycle (k_t and k_s, respectively). Note that training the teacher and the student until convergence for three cycles resembles the noisy student (NS) procedure [65]. In Fig. 4a we compare the performance of the best MILe iteration schedule with the NS schedule. We found that MILe achieves the best performance in terms of the ReaL-F1 score. In App. A.5 we ablate the pseudo label threshold (ρ) value.

We also investigate the effect of the number of teacher iterations (k_t) and student iterations (k_s) per cycle on the final performance (Fig. 4b, 4c). We report the ReaL-F1 for different k_t values (rows) and k_s values (columns). In Fig. 4b, we report the performance when training with 10% of the data, and in Fig. 4c we report the performance when training with all the data. In general, we find that 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, being the model 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.

5 Conclusions

We introduced multi-label iterated learning (MILe) to address the problem of label ambiguity and label noise in popular classification datasets such as ImageNet. MILe relaxes the single-label 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 single-label noisy predictions. MILe improves the performance of image classifiers for the single-label and multi-label problems, domain generalization, semi-supervised learning, and continual learning on IIRC. A possible limitation of iterated learning is choosing the correct teacher (k_t) and student iterations (k_s). However, this is an inherent problem of iterated learning [45]. In addition, our ablation experiments suggest that the proposed procedure is beneficial for a wide range of k_t and k_s values (Sec. 4.5). We hope that our research will open new avenues for iterated learning in the visual domain.

6 Reproducibility Statement

We have presented MILe, a multi-label iterated learning procedure to alleviate the label ambiguity problem in singly-labeled datasets such as ImageNet [18]. An overview of our method is depicted in Fig. 1, and details are provided in Alg. 1. We describe datasets, baselines, and metrics used for the experiments in each of their corresponding sections. We provide additional details about the metrics used in IIRC in App. A.3. We provide ablation experiments in Sec. 4.5. We provide the code in the supplementary material, and we will make it publicly available on Github.

References

[1] Mohamed Abdelsalam, Mojtaba Faramarzi, Shagun Sodhani, and Sarath Chandar. Iirc: Incremental implicitly-refined classification. *CVPR*, 2021.

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

[3] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[4] Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *ICML*, 2017.

[5] Lei Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? *arXiv preprint arXiv:1312.6184*, 2013.

[6] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Joshua B. Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. In *NeurIPS*, 2019.

[7] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 456–473, 2018.

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

[9] Lucas Beyer, Olivier J Hénaff, Alexander Kolesnikov, Xiaohua Zhai, and Aäron van den Oord. Are we done with imagenet? *arXiv preprint arXiv:2006.07159*, 2020.

[10] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD*, pp. 535–541, 2006.

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

[12] João Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. *CVPR*, 2017.

[13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pp. 1597–1607, 2020.

[14] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint:2006.10029*, 2020.

[15] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

[16] Michael Cogswell, Jiasen Lu, Stefan Lee, Devi Parikh, and Dhruv Batra. Emergence of compositional language with deep generational transmission. *arXiv preprint arXiv:1904.09067*, 2019.

[17] Gautier Dagan, Dieuwke Hupkes, and Elia Bruni. Co-evolution of language and agents in referential games. *arXiv preprint arXiv:2001.03361*, 2020.

[18] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

[19] Tommaso Furlanello, Zachary Lipton, Michael Tschannen, Laurent Itti, and Anima Anandkumar. Born again neural networks. In *ICML*, 2018.

[20] Priya Goyal, Mathilde Caron, Benjamin Lefaudeux, Min Xu, Pengchao Wang, Vivek Pai, Mannat Singh, Vitaliy Liptchinsky, Ishan Misra, Armand Joulin, et al. Self-supervised pretraining of visual features in the wild. *arXiv preprint arXiv:2103.01988*, 2021.

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

[22] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *NeurIPS*, 2020.

[23] S. Guo, Weilin Huang, H. Zhang, Chenfan Zhuang, Dengke Dong, M. Scott, and Dinglong Huang. Curriculumnet: Weakly supervised learning from large-scale web images. In *ECCV*, 2018.

[24] Shangmin Guo, Yi Ren, Serhii Havrylov, Stella Frank, Ivan Titov, and Kenny Smith. The emergence of compositional languages for numeric concepts through iterated learning in neural agents, 2019.

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

[26] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask r-cnn. *ICCV*, 2017.

[27] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019.

[28] Tong He, Zhi Zhang, Hang Zhang, Zhongyue Zhang, Junyuan Xie, and Mu Li. Bag of tricks for image classification with convolutional neural networks. In *CVPR*, pp. 558–567, 2019.

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

[30] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *ICML*, 2018.

[31] Simon 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.

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

[33] Simon Kirby, Tom Griffiths, and Kenny Smith. Iterated learning and the evolution of language. *Current opinion in neurobiology, 28:108–114*, 2014.

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

[35] David Krueger, Ethan Caballero, Jörn-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Rémi Le Priol, and Aaron C. Courville. Out-of-distribution generalization via risk extrapolation (rex). *CoRR*, 2020.

[36] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010. URL http://yann.lecun.com/exdb/mnist/.

[37] Kuang-Huei Lee, Xiaodong He, Lei Zhang, and Linjun Yang. Cleannet: Transfer learning for scalable image classifier training with label noise. In *CVPR*, 2018.

[38] Fushan Li and Michael Bowling. Ease-of-teaching and language structure from emergent communication. In *NeurIPS*, 2019.

[39] Junnan Li, Caiming Xiong, and Steven CH Hoi. Mopro: Webly supervised learning with momentum prototypes. *ICLR*, 2021.

[40] Wen Li, Limin Wang, Wei Li, E. Agustsson, and L. Gool. Webvision database: Visual learning and understanding from web data. *ArXiv*, 2017.

[41] Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang. Selective kernel networks. In *CVPR*, pp. 510–519, 2019.

[42] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *NeurIPS*, 2020.

[43] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *ICCV*, 2015.

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

[45] Yuchen Lu, Soumye Singhal, Florian Strub, Aaron Courville, and Olivier Pietquin. Countering language drift with seeded iterated learning. In *ICML*, 2020.

[46] Yuchen Lu, Soumye Singhal, Florian Strub, Olivier Pietquin, and Aaron Courville. Supervised seeded iterated learning for interactive language learning. In *EMNLP*, 2020.

[47] Marc Masana, Xialei Liu, Bartlomiej Twardowski, Mikel Menta, Andrew D Bagdanov, and Joost van de Weijer. Class-incremental learning: survey and performance evaluation. *arXiv preprint arXiv:2010.15277*, 2020.

[48] Hossein Mobahi, Mehrdad Farajtabar, and Peter L. Bartlett. Self-distillation amplifies regularization in hilbert space, 2020.

[49] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *CoRR*, 2021.

[50] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? *arXiv*, 2019.

[51] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *ICML*, 2019.

[52] Yi Ren, Shangmin Guo, Matthieu Labeau, Shay B. Cohen, and Simon Kirby. Compositional languages emerge in a neural iterated learning model. In *ICLR*, 2020.

[53] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, pp. 211–252, 2015.

[54] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. 2015.

[55] Manan Shah, Krishnamurthy Viswanathan, Chun-Ta Lu, Ariel Fuxman, Zhen Li, Aleksei Timofeev, Chao Jia, and Chen Sun. Inferring context from pixels for multimodal image classification. In *CIKM*. ACM, 2019. ISBN 9781450369763.

[56] Vaishaal Shankar, Rebecca Roelofs, Horia Mania, Alex Fang, Benjamin Recht, and Ludwig Schmidt. Evaluating machine accuracy on imagenet. In *ICML*, 2020.

[57] Jeremy Speth and Emily M Hand. Automated label noise identification for facial attribute recognition. In *CVPR Workshops*, pp. 25–28, 2019.

[58] Pierre Stock and Moustapha Cisse. Convnets and imagenet beyond accuracy: Understanding mistakes and uncovering biases. In *ECCV*, pp. 498–512, 2018.

[59] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *ICCV*, 2017.

[60] D. Tsipras, Shibani Santurkar, Logan Engstrom, Andrew Ilyas, and A. Madry. From imagenet to image classification: Contextualizing progress on benchmarks. In *ICML*, 2020.

[61] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. From imagenet to image classification: Contextualizing progress on benchmarks. In *ICML*, 2020.

[62] Yi Tu, Li Niu, Dawei Cheng, and Liqing Zhang. Protonet: Learning from web data with memory. *CVPR*, 2020.

[63] Grant Van Horn, Steve Branson, Ryan Farrell, Scott Haber, Jessie Barry, Panos Ipeirotis, Pietro Perona, and Serge Belongie. Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection. In *CVPR*, 2015.

[64] Ankit Vani, Max Schwarzer, Yuchen Lu, Eeshan Dhekane, and Aaron Courville. Iterated learning for emergent systematicity in vqa. In *ICLR*, 2021.

[65] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *CVPR*, pp. 10687–10698, 2020.

[66] Jingkang Yang, Litong Feng, Weirong Chen, Xiaopeng Yan, Huabin Zheng, Ping Luo, and Wayne Zhang. Webly supervised image classification with self-contained confidence. *ECCV*, 2020.

[67] Sangdoo Yun, Seong Joon Oh, Byeongho Heo, Dongyoon Han, Junsuk Choe, and Sanghyuk Chun. Re-labeling imagenet: from single to multi-labels, from global to localized labels. In *CVPR*, 2021.

[68] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3): 107–115, 2021.

[69] Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In *ICCV*, pp. 3713–3722, 2019.

[70] Ning Zhang, Manohar Paluri, Marc'Aurelio Ranzato, Trevor Darrell, and Lubomir Bourdev. Panda: Pose aligned networks for deep attribute modeling. In *CVPR*, 2014.

[71] Zizhao Zhang, Han Zhang, Sercan O Arik, Honglak Lee, and Tomas Pfister. Distilling effective supervision from severe label noise. In *CVPR*, 2020.

[72] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017.

A APPENDIX

We provide additional experiment on a synthetic setup along with results to assess the fraction of ImageNet-ReaL labels recovered with MILe. We then provide details on the metrics used for evaluation on the IRCC benchmark [1] (Sec. A.3). Fianlly, we provide additional results for multi-label classification on CelebA (Sec. A.6).

A.1 MULTI-LABEL 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.

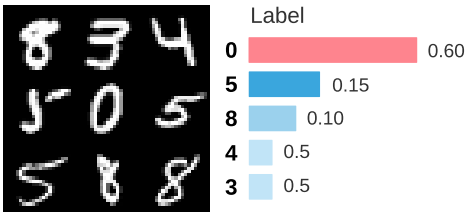


Figure 6: **Multi-MNIST.** The center digit has a probability of 0.6 to be chosen as the label for the whole grid.

Sec. 4.1 explores whether the multi-label representations built by MILe help to alleviate the label ambiguity problem in the ImageNet. However, many images in the 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 design 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 digits [36]. 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. 6). 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.

A.2 REAL LABEL RECOVERY

The goal of MILe is to alleviate the problem of label ambiguity by recovering all the alternative labels for a given sample. We define alternative labels as those that were not originally present in the ground truth. In this section, we evaluate how much of those alternative labels are recovered with MILe.

Method	ResNet-50		ResNet-18	
	10% data	100% data	10% data	100% data
Softmax	0.2171	0.2679	0.1983	0.2648
Sigmoid	0.2310	0.2845	0.2047	0.2836
MILe (ours)	**0.3042**	**0.3248**	**0.2187**	**0.2880**

Table 6: **Secondary label recovery.** Mean average precision over labels that appear in ReaL but not in the original ImageNet validation set.

Table 6 displays the mean average precision on the alternative labels present in ReaL [9]. As it can be seen, MILe is able to recover up to 7% more labels than replacing softmax by sigmoid and binary cross entropy during training.

A.3 IIRC BENCHMARK METRICS

In this section we detail the metrics used to report results for the IIRC benchmark. As it can be seen in Fig. 3, the two reported metrics are the precision-weighted Jaccard similarity and the mean precision-weighted Jaccard similarity.

Precision-weighted Jaccard Similarity. The Jaccard similarity (JS) refers to the intersection over union between model predictions $\hat{Y}_i$ and ground truth Y_i for the ith sample:

$$JS = \frac{1}{n} \sum_{i=1}^{n} \frac{|Y_i \cap \hat{Y}_i|}{|Y_i \cup \hat{Y}_i|}, \tag{2}$$

The precision-weighted JS for task k is the product between the JS and the precision for the samples belonging to that task:

$$R_{jk} = \frac{1}{n_k} \sum_{i=1}^{n_k} \frac{|Y_{ik} \cap \hat{Y}_{ik}|}{|Y_{ik} \cup \hat{Y}_{ik}|} \times \frac{|Y_{ik} \cap \hat{Y}_{ik}|}{\hat{Y}_{ik}}$$

where $(j \geq k), \hat{Y}_{ik}$ is the set of (model) predictions for the ith sample in the kth task, Y_{ik} are the ground truth labels, and n_k is number of samples in the task. R_{jk} can be used as a proxy for the model's performance on the kth task as it trains on more tasks (i.e. as j increases).

Mean precision-weighted Jaccard similarity. We evaluate the overall performance of the model after training until the task j, as the average precision-weighted Jaccard similarity over all the classes that the model has encountered so far. Note that during this evaluation, the model has to predict all the correct labels for a given sample, even if the labels were seen across different tasks.

A.4 DOMAIN GENERALIZATION

Figure 7: **ColoredMNIST+.** During training, the model is asked to classifier either digits or colors. Digits are highly correlated with their color, e.g. 0-4 tend to be green while 5-9 tend to be red. At test time, digits are less correlated with color.

In order to investigate how models perform outside of their original training distribution, Arjovsky et al. [3] introduced ColoredMNIST, a dataset of digits presented in different colors. In order to create spurious correlations, the color of the digits is highly correlated with the value itself. During training, data is sampled from two different image-label distributions or environments. In the first one, the correlation between digit and color is 90% and in the second is 80%. The correlation between the digit and color is 10% at test time. Since we want to explore the effect on generalization when the model is able to predict the digit and the color independently, we add a 33% chance of showing a blank image with no digit and only background color, where the background color is the label. This would be equivalent to a "beach" class in ImageNet. Note that this change does not remove the spurious correlations between the existing digits and their color. We call this benchmark ColoredMNIST+, see Fig. 7. During training, iterated learning builds a multi-label represenation of the digits, often including their color, leading to better disentanglement of the concepts "digits" and "color".

A.5 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). Table 7 shows the validation accuracies and ReaL-F1 scores for different threshold values when trained with 10% of

Threshold ρ	**ReaL-F1**	**Accuracy**
0.25	58.9	63.2
0.5	58.5	62.9
0.6	57.8	62.3
0.75	53.1	59.3
0.95	48.6	52.3

Table 7: **Pseudo label threshold ablation study.** ReaL F-1 and accuracy scores for a threshold value sweep (ρ). This experiment was conducted on ImageNet with the 10% data fraction setting.

Method	F1-score
CE-Sigmoid	80.14
ResNet-18(FPR) [8]	77.55
ResNet-34 (FPR) [8]	79.96
MILe (ours)	**81.40**

Table 8: Comparison on CelebA multi-attribute classification. Just as in ReaL ImageNet validation, we use F1-score (based on the intersection over union) measure to evaluate the methods.

	5 o Clock Shadow	*Arched Eyebrows*	*Attractive*	*Bags Under Eyes*	*Bald*	*Bangs*	*Big Lips*	*Big Nose*	*Black Hair*	*Blond Hair*	*Blurry*	*Brown Hair*	*Bushy Eyebrows*	*Chubby*	*Double Chin*	*Eyeglasses*	*Goatee*	*Gray Hair*	*Heavy Makeup*	*High Cheekbones*
Triplet-kNN [54]	66	73	83	63	75	81	55	68	82	81	43	76	68	64	60	82	73	72	88	86
PANDA [70]	76	77	85	67	74	92	56	72	84	91	50	**85**	74	65	64	88	84	79	95	**89**
Anet [43]	81	76	**87**	70	73	90	57	**78**	**90**	90	56	83	82	70	68	95	**86**	85	**96**	**89**
MILe	**85**	**83**	82	**74**	**82**	**92**	**65**	74	88	**91**	**76**	79	**83**	**72**	**72**	**98**	**86**	**86**	93	**89**

	Male	*Mouth Slightly Open*	*Mustache*	*Narrow Eyes*	*No Beard*	*Oval Face*	*Pale Skin*	*Pointy Nose*	*Receding Hairline*	*Rosy Cheeks*	*Sideburns*	*Smiling*	*Straight Hair*	*Wavy Hair*	*Wearing Earrings*	*Wearing Hat*	*Wearing Lipstick*	*Wearing Necklace*	*Wearing Necktie*	*Young*
Triplet-kNN [54]	91	92	57	47	82	61	63	61	60	64	71	92	63	77	69	84	91	50	73	75
PANDA [70]	**99**	93	63	51	87	66	69	67	67	68	81	**98**	66	78	77	90	**97**	51	**85**	78
Anet [43]	**99**	**96**	61	57	93	**67**	**77**	69	70	**76**	79	97	69	81	83	90	95	**59**	79	**84**
MILe	**99**	95	**74**	**77**	**94**	64	75	**69**	**77**	74	**87**	94	**74**	**83**	**84**	**94**	93	56	77	81

Table 9: Mean per-class balanced accuracy in percentage points for each of the 40 face attributes on CelebA.

the ImageNet data. Lower thresholds bias the student towards producing multi-label outputs, even for low-confidence classes. Larger threshold values makes the student tend towards single-label 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.

A.6 Multi-label classification on CelebA

We provide results on CelebA [43], a multi-label dataset. CelebA is a large-scale dataset of facial attributes with more than 200K celebrity images, each with 40 attribute annotations that are known to be noisy [57]. We report results in Table 8. Interestingly, despite the fact that CelebA is a multi-label dataset, we observe a $\sim 1\%$ improvement in F1 score when using the proposed iterative learning procedure. This along with per-class balanced accuracy in Table 9 is in line with our hypothesis that the iterated learning bottleneck has a regularization effect that prevents the model from learning noisy labels [45]. It is worth noting that MILe shows improved scores for the attributes that are difficult to classify such as *big-lips*, *arched-eyebrows* and *moustache*.