Supplementary Material

In Section A, we provide continual learning results on the IRCC benchmark [1]. In Section B we investigate to which extent MILe is able to recover labels that were not present in the original dataset. In Section C we provide additional details on the domain generalization experiment. In Section D, we provide additional results for multi-label classification on CelebA. In Section E, we test additional iterated learning schedules such as that of noisy student.

A. 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 [45] where the incoming batches of classes have two granularity levels, e.g. a coarse and a fine label. Labels are 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.

A.1. Metrics

As it can be seen in Fig. 6, 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:

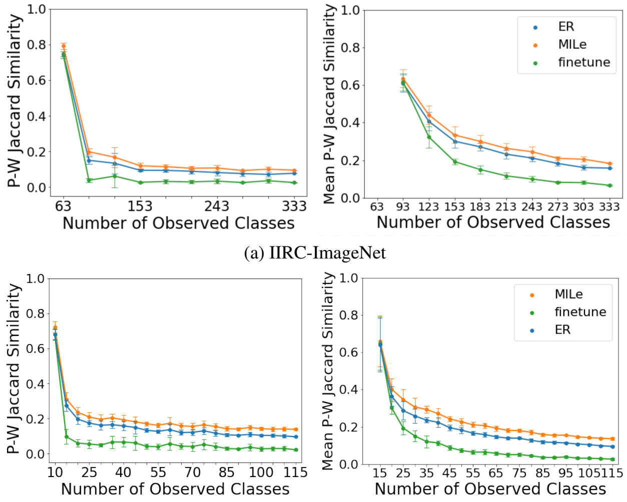
$$JS = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap \hat{Y}_i|}{|Y_i \cup \hat{Y}_i|}, \tag{2}$$

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) IIRC-ImageNet

(b) IIRC-CIFAR10

Figure 6. **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.

A.2. 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. We report IIRC-ImageNet-lite evaluation scores in Fig. 6a and CIFAR in Fig. 6b. 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.

B. 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 [8]. 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.

C. Details on Domain Generalization

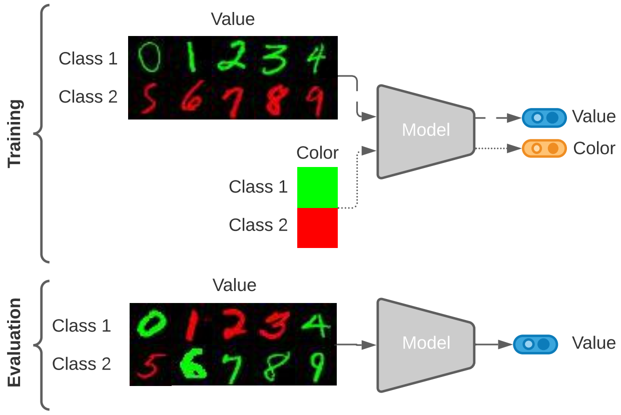


Figure 7. **ColoredMNIST+.** During training, the model is asked to classifier either digits or colors. Digits are highly correlated with their color, e.g. 0-4 tend to be green while 5-9 tend to be red. At test time, digits are less correlated with color.

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".

Method	F1-score
CE-Sigmoid	80.14
ResNet-18(FPR) [7]	77.55
ResNet-34 (FPR) [7]	79.96
MILe (ours)	**81.40**

Table 7. 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.

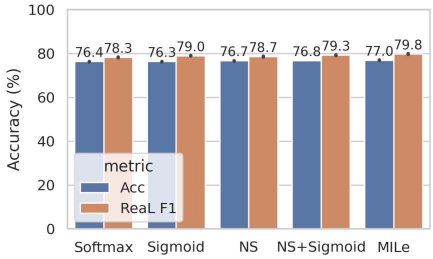


Figure 8. **Ablation study.** Comparison with noisy student (NS).

D. Multi-label classification on CelebA

We provide results on CelebA [41], 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 [55]. We report results in Table 7. 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 8 is in line with our hypothesis that the iterated learning bottleneck has a regularization effect that prevents the model from learning noisy labels [43]. 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*.

E. Comparisons with Noisy Student Scehduling

Xie et al. [63] introduced noisy student for labeling unlabeled data during semi-supervised learning. This is different from the goal of MILe, which is to construct a new multi-label representation of the images from single labels. Dif-

	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 [52]	66	73	83	63	75	81	55	68	82	81	43	76	68	64	60	82	73	72	88	86
PANDA [68]	76	77	85	67	74	92	56	72	84	91	50	**85**	74	65	64	88	84	79	95	**89**
Anet [41]	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 [52]	91	92	57	47	82	61	63	61	60	64	71	92	63	77	69	84	91	50	73	75
PANDA [68]	**99**	93	63	51	87	66	69	67	67	68	81	**98**	66	78	77	90	**97**	51	**85**	78
Anet [41]	**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 8. Mean per-class balanced accuracy in percentage points for each of the 40 face attributes on CelebA.

ferent from MILe, which trains a succession of short-lived teacher and student models, noisy student trains the model three times until convergence. This raises the question of how would MILe perform if it followed noisy student's iteration schedule instead of the one introduced in the main text.

In Fig. 8 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.