Semi-Supervised Learning with Taxonomic Labels

Jong-Chyi Su
jcsu@cs.umass.edu

Subhransu Maji
smaji@cs.umass.edu

University of Massachusetts Amherst
Amherst, MA, USA

Abstract

We propose techniques to incorporate coarse taxonomic labels to train image classi-fiers in fine-grained domains. Such labels can often be obtained with a smaller effort for fine-grained domains such as the natural world where categories are organized accord-ing to a biological taxonomy. On the Semi-iNat dataset consisting of 810 species across three Kingdoms, incorporating Phylum labels improves the Species level classification accuracy by 6% in a transfer learning setting using ImageNet pre-trained models. Incor-porating the hierarchical label structure with a state-of-the-art semi-supervised learning algorithm called FixMatch improves the performance further by 1.3%. The relative gains are larger when detailed labels such as Class or Order are provided, or when models are trained from scratch. However, we find that most methods are not robust to the presence of out-of-domain data from novel classes. We propose a technique to select relevant data from a large collection of unlabeled images guided by the hierarchy which improves the robustness. Overall, our experiments show that semi-supervised learning with coarse taxonomic labels are practical for training classifiers in fine-grained domains.

1 Introduction

Large labeled datasets have been the key to the success of deep networks for many tasks. However, labeling requires expertise and can be time-consuming, especially for fine-grained recognition tasks such as identifying the species of birds [44] or variants of aircrafts [74]. In this work, we investigate if coarsely labeled datasets can be used to improve performance of a target fine-grained recognition task. For example, in natural domains one can often obtain a large dataset with the same coarse labels as the target task through community driven plat-forms such as iNaturalist [43]. Effectively incorporating them in a semi-supervised learning framework to improve performance could be a compelling alternative to existing few-shot learning approaches, which have been less effective in fine-grained domains [10, 40].

We present an analysis on the Semi-iNat dataset [48] that consists of images from 810 species spanning three Kingdoms and eight Phyla (Figure 1 and Table 1). The dataset con-tains: (i) a small set of images labeled at the species level (in-class), (ii) a large set (9×) of coarsely-labeled images from the same species (in-class), and (iii) an even larger set (32×) of coarsely-labeled images from novel species within the same taxonomy (out-of-class). At test
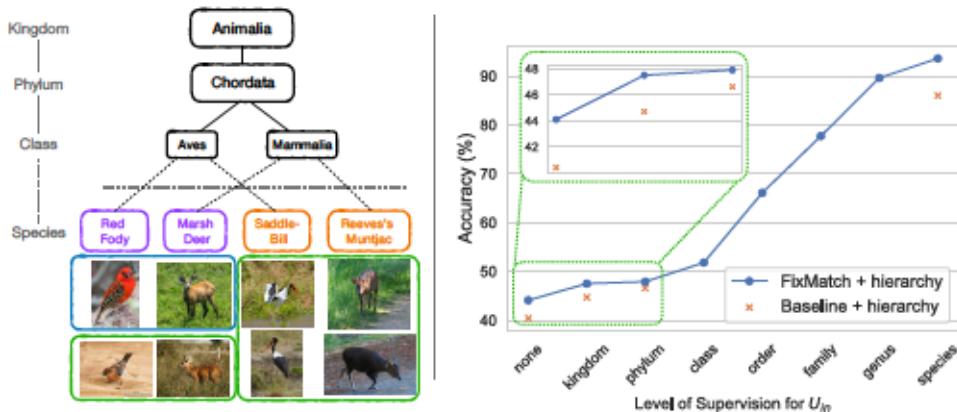
Figure 1: **Taxonomic labels improve semi-supervised learning. Left:** The Semi-iNat dataset [36] contains labeled data from in-class Species and coarsely labeled data from both in-class and out-of-class Species. The coarse labels such as Kingdom and Phylum, can be derived from the taxonomy given the Species, but are easier to annotate. **Right:** We observe that incorporating coarse labels improves the Baseline and FixMatch across different levels of supervision. FixMatch provides additional gains over the supervised Baseline which is based on fine-tuning an ImageNet pre-trained ResNet-50 using a hierarchy-aware loss (§ 3).

time, the species classification accuracy is measured on novel images of the set of species within the labeled set (in-class). The dataset is fine-grained and long-tailed, posing challenges to existing approaches for semi-supervised learning.

For a consistent evaluation, we present results using a ResNet-50 network trained from scratch or pre-trained on ImageNet [17] using various semi-supervised and self-supervised learning approaches. For the supervised Baseline with ImageNet pre-training, incorporating a hierarchical loss at the Phylum level consisting of eight categories improves the Top-1 accuracy from 40.4% to 46.6% (Figure 1 and Table 2). This beats the gains using semi-supervised learning with FixMatch [37] alone, which obtains 44.1%. However, the gains are complementary and combining the two improves the performance to 47.9%. Coarse taxonomic labels are also useful when models are trained from scratch, improving the performance of the best method from 32.0% to 34.5% (Table 2). Figure 1 quantifies the gains obtained with the Baseline and FixMatch using supervision from the Kingdom level (3 categories) to the Species level (810 categories, full supervision). Kingdom and Phylum supervision provides consistent gains for both methods, and improves over the semi-supervised learning without the coarse labels. The benchmark is far from saturated as indicated by the performance of the fully-supervised Baseline at roughly 86%.

A common assumption in semi-supervised learning is that the weakly-labeled data belongs to the same set of classes as the target task. This is hard to guarantee in practice as coarsely labeled datasets collected in-the-wild may contain novel classes. We find that the presence of this out-of-domain data leads to a drop in performance for nearly all approaches. For example, the performance of FixMatch drops from 47.9% to 41.1% (Table 3) when the larger set of images are included for semi-supervised learning. This is problematic because labeling if an image contains novel classes is significantly more challenging than obtaining a large pool of images with the same coarse labels. We find that prior work based on im-

portance weighting based on a domain classifier (*e.g.*, [39]) is less effective, but using the hierarchy to exclude the novel categories leads to a small improvement in some cases.

In summary, we show that coarse labels improve supervised and semi-supervised learning in fine-grained natural taxonomies. In particular, we find that: (i) coarse labels can be incorporated in several state-of-the-art methods to boost their performance; (ii) The improvements are greater with more fine-grained labels; (iii) The presence of novel classes hurts performance, but this can be somewhat mitigated by techniques for detecting domain shifts. However, the marginal gains obtained in this setting highlights the difficulty of detecting novel classes in fine-grained domains. The code is available at https://github.com/cvl-umass/ssl-evaluation.

2 Related Works

Semi-supervised learning. Semi-supervised learning aims to use weakly labeled data to improve the model generalization. Self-training approaches proposed in some early work [25, 35] use the model's own prediction to generate labels. Their modern incarnations include pseudo-labeling [22] which uses confident predictions as target labels for unlabeled data. Such labels can also be added gradually as a form of curriculum learning [2, 6] to reduce model drift. UPS [32] generalizes this idea and incorporates low-probability predictions as negative pseudo-labels for multi-label prediction tasks. Other methods use a combination of self-supervised and semi-supervised learning techniques [14, 39, 50], which is sometimes followed by an additional step where the model's predictions are used to train a "student model" using distillation [8, 47, 48, 53]. Consistency-based approaches enforce the similarity of predictions between two augmentations of the same data, or different models in an ensemble, as a form of regularization [1, 21, 26, 29, 43]. The role of augmentations have been explored in detail in techniques such as MixMatch [3], ReMixMatch [4], FixMatch [37], and UDA [46], which combine geometric and photometric image augmentations with other techniques such as MixUp [51].

Learning with hierarchical labels. The hierarchical structure of the label space can be used to improve classification performance in many ways [16, 31, 41]. A common approach is to frame the problem as a structured prediction task and use a graphical model that incorporates the label structure with standard Bayesian machinery. For example, Deng *et al.* [13] exploit the inclusion and exclusion relations among the labels to improve classification. YOLOv2 [30] learns object detectors across a large set of categories by predicting labels on the taxonomy in a top-down manner by learning a conditional distribution of the leaves given the parent category. This approach parameterizes the learnable weights along the edges of the tree. We instead parameterize the weights along the leaves of the tree. Other works predict fine-grained labels by designing models that predict the labels [49, 52] or learn features [45] across different levels in the hierarchy in a multi-task framework. The hierarchical label space can also be utilized to detect novel classes. For example, given coarse labels, Hsieh *et al.* [20] learn to assign fine-grained pseudo-labels by meta-learning, assuming that the classifier can achieve the best performance when the missing labels are correctly recovered. Another application is zero-shot learning where attributes of the novel classes are provided, but there are no training images in novel classes [23, 44]. Recently, coarse labels have been incorporated in contrastive learning to improve image retrieval [17] and few-shot learning [5, 28]. Unlike prior work, we investigate if hierarchical labels can be

used for improving of semi-supervised learning, for example by constraining the label space of approaches such as FixMatch or Pseudo-Labeling.

3 Method

Notation and problem setting. We focus on a structured prediction task where the label space $y \in \mathcal{Y}$ has a hierarchical structure. It corresponds to a tree-structured biological taxonomy with 7 levels corresponding to the *Kingdom, Phylum, Class, Order, Family, Genus,* and *Species*. Denote y^l as the label of an instance at the level l. Thus, y^1 is the label at the Kingdom level, y^2 for the label in the Phylum level, and the leaf nodes in the tree correspond to Species-level labels denoted by y^7. Similarly, denote the sets of label space at a level l as C^l. For example, the label space in the species level is $y^7 \in C^7$, and in the Phylum level is $y^2 \in C^2$. Given the label in a level, we can infer the labels in all the upper levels using the tree structure, *e.g.*, we can infer the Kingdom label given the Phylum label. The Semi-iNat [58] dataset provides Species-level labels for a subset of images, but coarse labels (*e.g.*, Kingdom and Phylum) for a larger set of images (Table 1). Performance is measured as the accuracy at the Species level on novel images.

Model parameterization. We train a model to predict the labels at the finest level, *i.e.*, $y^7 = f(x)$, and marginalize over the tree structure to obtain probabilities at any level. There are several alternatives to this schema. For example, we could train separate heads for each level instead of just the leaves. However, this performed worse, obtaining 42.1% accuracy on a supervised model trained from ImageNet, while our proposed method achieves 46.6% accuracy. We could also consider the parameterization proposed in YOLOv2 [40] which models the conditional distribution of the leaves given the parent for each internal node in the tree. Both schemes require many more parameters. For example, there are 2041 edges in the taxonomy (summing over Table 1 right); thus, the edge-parameterization of YOLOv2 would require 2041 weights (#edges) compared to 810 weights (#leaves). While the edge parameterization can handle arbitrary graphs, it offers no obvious advantage for tree-structured models. Having fewer weights may be preferable in the few-shot setting.

3.1 Hierarchical supervised loss

We consider the supervised cross-entropy loss in each level of the hierarchy. For labeled data $(x_i, y_i^7) \in \mathcal{L}$, the model f first predicts the label space of the species C^7 with the probabilities $p_i^7 = f(x_i)$. For coarse labeled data, say we have the label at the Phylum level $(u_j, y_j^2) \in \mathcal{U}$, we first use the same model to predict the labels in the species level $q_j^7 = f(u_j)$. We then apply a cross-entropy loss over the model's prediction at the Phylum level obtained by summing the probabilities of all the leaf nodes under each Phylum. The marginalization can be done by $q_j^2 = q_j^7 \cdot W_7^2$, where the predefined matrix W_7^2 represents the edges between the Species and Phylum level (the elements are 1 for the edges and 0 otherwise).

During training, we sample m labeled data from $\mathcal{L}$ and n coarsely labeled data from $\mathcal{U}$ in each batch for stochastic gradient descent. For labeled data, we only add supervised loss on the lowest level, which is the Species level. The complete hierarchical supervised loss is:

$$\mathcal{L}_{\text{hie}}^{7,2} = \sum_{i=1}^{m} H(y_i^7, p_i^7) + \sum_{j=1}^{n} H(y_j^2, q_j^2), \qquad (1)$$

where H is the cross-entropy function $H(u,v) = -\sum_w u(w)\log v(w)$. The first term is the loss for labeled data on the Species level, and the second term is the loss for coarsely labeled data on the Phylum level. The superscript on the loss $\mathcal{L}_{\text{hie}}^{7,2}$ represents the level of supervision for labeled and coarsely labeled data. In the ablation studies, we will investigate the effect of using different levels of supervision. Note that we can add supervised loss on all seven levels of the taxonomy. However, in our experiments we did not find it useful to add losses at a level coarser than which the supervision was provided, *e.g.* losses at the Genus level or higher when Species labels are provided. Our method can also be extended to general hierarchical graphs such as WordNet using marginalization methods [13, 14].

3.2 Joint training with semi-supervised loss

In addition to the hierarchical loss, we add semi-supervised losses such as consistency regularization, entropy minimization, or pseudo-labeling on the Species level for coarsely labeled data. We select representative semi-supervised methods including Pseudo-Label, FixMatch, Self-Training with distillation, and Self-Supervised learning (MoCo) with distillation. We describe how we incorporate hierarchical supervisions with these methods in the following.

Pseudo-Label [22]. Pseudo-label uses the model's predictions as labels if the prediction is higher than a threshold τ. Denote the pseudo-label in the Species level as $\hat{q}_i^7 = \arg\max(q_i^7)$, then the loss for pseudo-label training is:

$$\mathcal{L} = \mathcal{L}_{\text{hie}}^{7,2} + \sum_{j=1}^{n} \mathbb{1}\left[\max(r_i) \geq \tau\right] H(\hat{q}_i^7, q_i^7). \tag{2}$$

FixMatch [47]. FixMatch utilized two different augmentation functions for consistency training, one with weak augmentation $\alpha(\cdot)$ and one with strong augmentation $\mathcal{A}(\cdot)$. For each coarsely labeled image u_j, the KL distance between the pseudo-label from weakly-augmented image $q_j = f(\alpha(u_j))$ and the prediction of strongly-augmented image $Q_j = f(\mathcal{A}(u_j))$ is minimized. To compute the supervised loss for labeled data, weak augmentation $p_i = f(\alpha(x_i))$ is used. For the supervised loss on coarsely labeled data $\mathcal{U}$, we use strongly-augmented images $Q_j = f(\mathcal{A}(u_j))$ since weakly-augmented images are only used for generating pseudo-labels without back-propagation. The final loss is:

$$\mathcal{L} = \underbrace{\sum_{i=1}^{m} H(y_i^7, p_i^7) + \sum_{j=1}^{n} H(y_j^2, Q_j^2)}_{\mathcal{L}_{\text{hie}}^{7,2}} + \sum_{j=1}^{n} \mathbb{1}\left[\max(q_j^7) \geq \tau\right] H(\hat{q}_i^7, Q_i^7). \tag{3}$$

Self-Training. We use model distillation [19] as the self-training method. Specifically, we first train a teacher model using labeled data for supervised learning $\mathcal{L} = \sum_{i=1}^{m} H(y_i^7, p_i^7)$. We then train a student model using distillation loss, which is the KL distance between the *logits* from the teacher and student models (denoted as z^t and z^s). The final loss is:

$$\mathcal{L} = \mathcal{L}_{\text{hie}}^{7,2} + \sum_{i=1}^{n} H\left(\sigma\left(\frac{z_i^t}{T}\right), \sigma\left(\frac{z_i^s}{T}\right)\right), \tag{4}$$

where $\sigma(\cdot)$ is the softmax function and T is the temperature parameter.

Self-Supervised Learning (MoCo) [18]. We use Momentum Contrastive (MoCo) [18] for self-supervised learning on the union of labeled and coarsely labeled data ($\mathcal{L} \cup \mathcal{U}$). Specifically, MoCo uses contrastive learning to minimize the representations of two different augmentations of an image. Denote $q = f(x)$ as the representation of an image x and k^+ as the positive sample, which is another augmentation of the same image x. The negative samples k_i^- are sampled from a memory bank. The InfoNCE [27] loss for the query q is:

$$\mathcal{L}_q = -\log \frac{\exp(q \cdot k^+ / T)}{\exp(q \cdot k^+ / T) + \sum_i^K \exp(q \cdot k_i^- / T)}, \qquad (5)$$

where T is a temperature hyper-parameter. The encoder for the memory bank is updated based on momentum of the encoder $f(\cdot)$ to stabilize the training. After the self-supervised pre-training is complete, we replace the MLP layers (after global pooling) with a linear projection layer and fine-tune the entire model using our supervised hierarchical loss (Eq. 1). Alternatives to MoCo such as SimCLR [7] and BYOL [15] could be used, but their effect is somewhat mitigated by the fact that ImageNet pre-training is still far more effective than self-supervised learning on fine-grained domains. However, the impact may be higher when training from scratch.

MoCo + Self-Training. This method combines the previous two methods, which is similar to the setting proposed in Chen *et al.* [8]. The MoCo pre-trained model is followed by supervised fine-tuning on the labeled set to obtain the teacher model. This is then used to self-train a student model using the distillation loss in Eq. 4.

4 Experiments

4.1 Experimental settings

Dataset. We use the Semi-iNat dataset [48] from the semi-supervised challenge [46] at the FGVC8 workshop. The dataset contains 810 in-class species (C_{in}) and 1629 out-of-class species (C_{out}) from 3 different Kingdoms. The fully labeled data come from in-class species while the coarsely labeled data are drawn from in-class and out-of-class species, denoted as U_{in} and U_{out} respectively. Table 1 shows the statistics of the dataset and Figure 1 left shows the taxonomy.

Training details. We use ResNet-50 [17] as our backbone model and an input size of 224×224 for all the experiments. For all the methods except for MoCo and FixMatch, we use SGD with a momentum of 0.9 to train the model with 100k (from scratch) or 50k iterations (from expert models). The batch size is 60 for training supervised baselines. For semi-supervised methods, we sample 30 images each from labeled and coarsely labeled data with a total batch size of 60. The learning rate is searched within $[0.001, 0.03]$, and the weight decay is set for either 0.001 or 0.0001. The hyper-parameters are set using the validation set of Semi-iNat. This set is not included for the supervised loss but is used for training MoCo.

For MoCo, we follow the setting of MoCov2 [9] and use a batch size of 2048 negative samples. The training is done with a learning rate of 0.03 and 0.0003, and for 800 and 200 epochs, for training from scratch and from expert models respectively.

For FixMatch, we follow the original setting and use RandAugment [11] for augmentation. Due to the hardware constraints, we were limited to a batch size of 32 for labeled data

Kingdom	Phylum	C_{in}	C_{out}
Animalia (1294)	Mollusca	11	24
	Chordata	113	228
	Arthropoda	301	605
	Echinodermata	4	8
Plantae (1028)	Tracheophyta	336	674
	Bryophyta	6	12
Fungi (117)	Basidiomycota	29	58
	Ascomycota	10	20
Total classes		810	2439

Taxonomy	#Classes in C_{in}
Kingdom	3
Phylum	8
Class	29
Order	123
Family	339
Genus	729
Species	810

Table 1: **Statistics of the Semi-iNat dataset [58]. Left: Number of classes under each Kingdom and Phylum.** The species of Semi-iNat come from 3 Kingdoms and 8 Phyla. In each Phylum, one-third of the species are used for in-class species C_{in} and the rest are used for out-of-class species C_{out}. **Right: Number of classes in each level of the taxonomy.**

	Method	from scratch		from ImageNet	
	(Phylum Supervision →)	w/o	w/	w/o	w/
U_{in}	Supervised Baseline	18.5	21.7↑	40.4	46.6↑
	Pseudo-Label [22]	18.6	22.7↑	40.3	44.9↑
	FixMatch [47]	15.5	25.7↑	44.1	47.9↑
	Self-Training	20.3	23.7↑	42.4	44.8↑
	MoCo [18]	30.2	33.5↑	41.7	41.9↑
	MoCo + Self-Training [40]	32.0	35.4↑	42.6	45.8↑

Table 2: **Results of adding Phylum supervision on the Semi-iNat dataset.** Adding hierarchical loss at the *Phylum* level improves the Baseline and all the Semi-Supervised methods when images come from U_{in}, *i.e.*, images with species labels corresponding to those in the test set C_{in}. The best methods are shown in teal.

and 160 for coarsely labeled data for training with 4 GPUs. When training from scratch we use a learning rate of 0.03 for 200k iterations; when training from expert models we use a learning rate of 0.001 for 100k iterations. The threshold for FixMatch is set as 0.8 for all our experiments.

4.2 Using Phylum level supervision

We first consider the setting where the coarsely labeled images are within the set of labeled species (U_{in}). In § 4.4 we will analyze the effect and utility of adding images of novel species from the same coarse categories. Models are initialized randomly or from an ImageNet pre-trained model. For each setting, the Baseline supervised learning and five Semi-Supervised
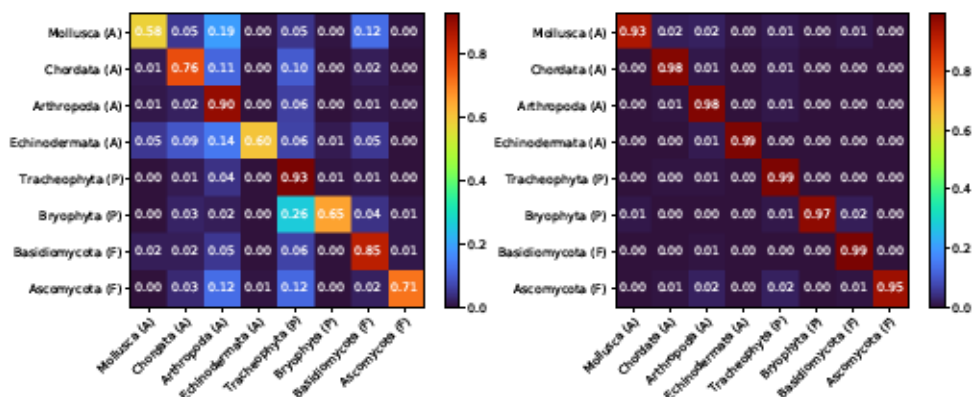
Figure 2: **Confusion matrices on the Phylum level. Left:** Supervised Baseline model *without* coarse-label supervision. **Right:** FixMatch + hierarchical loss on the Phylum level. Combining semi-supervised methods and hierarchical supervision on coarsely labeled data reduces the confusion between Phyla.

learning methods are evaluated. We then analyze the effect of adding hierarchical loss.

Results are presented in Table 2. Adding a hierarchical loss gives almost a 10% improvement for FixMatch and 3% for all other methods when models are trained from scratch. When initialized with ImageNet pre-trained models, adding hierarchical loss provides 2-6% improvements in top-1 accuracy except for the self-supervised training method (MoCo). Figure 2 shows the confusion matrices at the Phylum level for models trained with and without hierarchical supervision. Hierarchical supervision sensibly reduces confusion among the four Phyla within the animal (A) Kingdom (*e.g.*, Antropoda vs. Echinodermata), as well as at the Plant (P) Kingdom. The combined effect of hierarchical supervision and semi-supervised learning represents an overall improvement from 40.4% to 47.9%.

4.3   Using different levels of supervision

In this section, we consider supervision across different levels of the taxonomy on top of FixMatch and the supervised baseline. For example, if we have the labels in the Order level for coarsely labeled data, *i.e.* $(u_j, y_j^4) \in \mathcal{U}$, then the hierarchical loss becomes $\mathcal{L}_{\text{hie}}^{7,4}$. We use the ground-truth labels (on the Species level) of the coarsely labeled data which are released after the competition ends [56] for the analysis. As shown in Figure 1 right, we can see that finer-grained labels improve performance, but this also requires more annotation effort. The number of classes at each level in the taxonomy shown in Table 1 provides a proxy for the annotation effort. Even incorporating labels on the Kingdom level (which only has three categories) leads to improvements, while Class level (29 categories) improves the performance of FixMatch from 44.1% (FixMatch + None) to 51.8% (FixMatch + Class). In these experiments, the coarsely labeled set $\mathcal{U}_{in}$ on which the semi-supervised losses are constructed in Eq. 3 are kept the same. Interestingly, even when the method is fully supervised, these self-supervised losses are useful, and provide an improvement over the fully-supervised Baseline. This is perhaps not as surprising as previous works have noted that the self-supervised losses improve few-shot learning [14, 39, 50].

Method	from scratch		from ImageNet	
(Phylum Supervision →)	w/o	w/	w/o	w/
Supervised Baseline	18.5	20.5↑	40.4	45.6↑
Pseudo-Label [77]	18.8	21.2↑	40.3	44.0↑
FixMatch [47]	11.0	21.1↑	38.5	41.1↑
Self-Training	19.7	23.3↑	41.5	44.1↑
MoCo [18]	31.8	29.4↓	40.8	39.3↓
MoCo + Self-Training [40]	32.9	35.4↑	41.5	42.6↑

(row label at left, rotated: $U_{in} + U_{out}$)

Table 3: **Results on the Semi-iNat dataset with novel classes.** The presence of novel classes C_{out} when adding U_{out} reduces the performance of most methods compared to using U_{in} shown in Table 2. However, adding hierarchical loss provides improvements, both when training from scratch or from ImageNet. We also find that MoCo+Self-Training is the most robust when training from scratch, and no method is able to improve over the supervised Baseline when ImageNet pre-training is used. The best methods are shown in teal.

4.4 Effect of domain shift

Next, we consider the case when there is a domain shift, *i.e.* training with data from $U_{in} + U_{out}$. The results are shown in Table 3. When there is no hierarchical loss, the performances of semi-supervised methods all drop except for MoCo. Adding hierarchical loss improves all the methods except for MoCo, though the improvement is less compared to Table 2. Surprisingly, when training from the ImageNet model, no semi-supervised method improves over the supervised Baseline trained with the hierarchical loss. In particular, FixMatch is less robust to the domain shift, echoing the findings in [40]. One potential reason is the large domain shift between U_{in} and U_{out} the dataset: although the out-of-domain classes are drawn from the same Class in the taxonomy, the appearances can be significantly different.

To alleviate the effect of domain shift, we propose to filter the out-of-domain data by measuring the uncertainty in the model's predictions. This step is performed before any semi-supervised training. Similar to pseudo-labeling, we first use the baseline supervised model to generate predictions for images in $U_{in} + U_{out}$, then check if the maximum probability is greater than $\tau = 0.8$. Note that the set of out-of-domain classes (C_{out}) remains unknown. We further check if the predicted labels match the provided coarse labels at the Phylum level to filter out-of-domain images.

After filtering out-of-domain data, we train FixMatch using the selected coarse-labeled images. Using the supervision at the Phylum level obtains 42.0% accuracy compared to 41.1% without any domain selection. This allows us to use uncurated data, though the performance is lower than having only in-domain data (47.9%). Our proposed approach is simple, and there is significant room to improve this using better out-of-domain detection techniques and modeling the taxonomy. Nevertheless, the problem is challenging in fine-grained domains.

5   Conclusion

We showed that coarse labels improve semi-supervised learning on fine-grained image classification tasks. Thus, collecting coarse labels might provide a practical way to train models on new domains where a vast number of images are readily available, but labeling effort is expensive. We also found that out-of-domain (or novel class) data leads to a drop in performance. Hierarchical labels help, but the task of selecting relevant images can be challenging in fine-grained domains. Our model based on uncertainty and filtering using coarse labels only provides modest improvements. Improved techniques for detecting out-of-domain data combined with taxonomy-aware user input could provide further benefits.

References

[1] Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. In *Neural Information Processing Systems (NeurIPS)*, 2014.

[2] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *International Conference on Machine Learning (ICML)*, 2009.

[3] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Neural Information Processing Systems (NeurIPS)*, 2019.

[4] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring. *International Conference on Learning Representations (ICLR)*, 2020.

[5] Guy Bukchin, Eli Schwartz, Kate Saenko, Ori Shahar, Rogerio Feris, Raja Giryes, and Leonid Karlinsky. Fine-grained angular contrastive learning with coarse labels. In *Computer Vision and Pattern Recognition (CVPR)*, 2021.

[6] Paola Cascante-Bonilla, Fuwen Tan, Yanjun Qi, and Vicente Ordonez. Curriculum labeling: Self-paced pseudo-labeling for semi-supervised learning. *AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (ICML)*, 2020.

[8] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *Neural Information Processing Systems (NeurIPS)*, 2020.

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

[10] Elijah Cole, Xuan Yang, Kimberly Wilber, Oisin Mac Aodha, and Serge J. Belongie. When does contrastive visual representation learning work? *arXiv preprint arXiv:2105.05837*, 2021.

[11] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Neural Information Processing Systems (NeurIPS)*, 2020.

[12] 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*, pages 248–255. Ieee, 2009.

[13] Jia Deng, Nan Ding, Yangqing Jia, Andrea Frome, Kevin Murphy, Samy Bengio, Yuan Li, Hartmut Neven, and Hartwig Adam. Large-scale object classification using label relation graphs. In *European Conference on Computer Vision (ECCV)*, 2014.

[14] Spyros Gidaris, Andrei Bursuc, Nikos Komodakis, Patrick Pérez, and Matthieu Cord. Boosting few-shot visual learning with self-supervision. In *International Conference on Computer Vision (ICCV)*, 2019.

[15] 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. In *Neural Information Processing Systems (NeurIPS)*, 2020.

[16] Yanming Guo, Yu Liu, Erwin M Bakker, Yuanhao Guo, and Michael S Lew. Cnn-rnn: a large-scale hierarchical image classification framework. *Multimedia tools and applications*, 77(8):10251–10271, 2018.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision (ECCV)*, 2016.

[18] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

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

[20] Cheng-Yu Hsieh, Miao Xu, Gang Niu, Hsuan-Tien Lin, and Masashi Sugiyama. A pseudo-label method for coarse-to-fine multi-label learning with limited supervision. *ICLR LLD Workshop*, 2019.

[21] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. *International Conference on Learning Representations (ICLR)*, 2017.

[22] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, 2013.

[23] Kibok Lee, Kimin Lee, Kyle Min, Yuting Zhang, Jinwoo Shin, and Honglak Lee. Hierarchical novelty detection for visual object recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[24] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[25] Geoffrey J McLachlan. Iterative reclassification procedure for constructing an asymptotically optimal rule of allocation in discriminant analysis. *Journal of the American Statistical Association*, 70(350):365–369, 1975.

[26] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993, 2018.

[27] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[28] Cheng Perng Phoo and Bharath Hariharan. Coarsely-labeled data for better few-shot transfer. In *International Conference on Computer Vision (ICCV)*, 2021.

[29] Antti Rasmus, Mathias Berglund, Mikko Honkala, Harri Valpola, and Tapani Raiko. Semi-supervised learning with ladder networks. In *Neural Information Processing Systems (NeurIPS)*, 2015.

[30] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[31] Marko Ristin, Juergen Gall, Matthieu Guillaumin, and Luc Van Gool. From categories to subcategories: large-scale image classification with partial class label refinement. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[32] Mamshad Nayeem Rizve, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. In defense of pseudo-labeling: An uncertainty-aware pseudo-label selection framework for semi-supervised learning. *arXiv preprint arXiv:2101.06329*, 2021.

[33] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *Neural Information Processing Systems (NeurIPS)*, 2016.

[34] Colin Samplawski, Erik Learned-Miller, Heesung Kwon, and Benjamin M Marlin. Zero-shot learning in the presence of hierarchically coarsened labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 926–927, 2020.

[35] H Scudder. Probability of error of some adaptive pattern-recognition machines. *IEEE Transactions on Information Theory*, 11(3):363–371, 1965.

[36] Semi-iNat. *Semi-Supervised iNaturalist Challenge at FGVC8*. https://sites.google.com/view/fgvc8/competitions/semi-inat2021.

[37] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Neural Information Processing Systems (NeurIPS)*, 2020.

[38] Jong-Chyi Su and Subhransu Maji. The semi-supervised inaturalist challenge at the fgvc8 workshop. *arXiv preprint arXiv:2106.01364*, 2021.

[39] Jong-Chyi Su, Subhransu Maji, and Bharath Hariharan. When does self-supervision improve few-shot learning? In *European Conference on Computer Vision (ECCV)*, 2020.

[40] Jong-Chyi Su, Zezhou Cheng, and Subhransu Maji. A realistic evaluation of semi-supervised learning for fine-grained classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[41] Fariborz Taherkhani, Hadi Kazemi, Ali Dabouei, Jeremy Dawson, and Nasser M Nasrabadi. A weakly supervised fine label classifier enhanced by coarse supervision. In *International Conference on Computer Vision (ICCV)*, 2019.

[42] Hugo Touvron, Alexandre Sablayrolles, Matthijs Douze, Matthieu Cord, and Hervé Jégou. Grafit: Learning fine-grained image representations with coarse labels. *arXiv preprint arXiv:2011.12982*, 2020.

[43] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[44] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[45] Dequan Wang, Zhiqiang Shen, Jie Shao, Wei Zhang, Xiangyang Xue, and Zheng Zhang. Multiple granularity descriptors for fine-grained categorization. In *International Conference on Computer Vision (ICCV)*, 2015.

[46] Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. Unsupervised data augmentation for consistency training. *Neural Information Processing Systems (NeurIPS)*, 2020.

[47] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le. Self-training with noisy student improves imagenet classification. In *Computer Vision and Pattern Recognition (CVPR)*, 2020.

[48] I Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. Billion-scale semi-supervised learning for image classification. *arXiv preprint arXiv:1905.00546*, 2019.

[49] Zhicheng Yan, Hao Zhang, Robinson Piramuthu, Vignesh Jagadeesh, Dennis DeCoste, Wei Di, and Yizhou Yu. Hd-cnn: hierarchical deep convolutional neural networks for large scale visual recognition. In *International Conference on Computer Vision (ICCV)*, 2015.

[50] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4L: Self-supervised semi-supervised learning. *International Conference on Computer Vision (ICCV)*, 2019.

[51] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations (ICLR)*, 2018.

[52] Xinqi Zhu and Michael Bain. B-cnn: branch convolutional neural network for hierarchical classification. *arXiv preprint arXiv:1709.09890*, 2017.

[53] Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin D Cubuk, and Quoc V Le. Rethinking pre-training and self-training. *Neural Information Processing Systems (NeurIPS)*, 2020.