Learning Rare Category Classifiers on a Tight Labeling Budget

Ravi Teja Mullapudi[2,4] rmullapu@cs.cmu.edu
Fait Poms[1] fpoms@cs.stanford.edu
William R. Mark[4] billmark@google.com
Deva Ramanan[2,3] deva@cs.cmu.edu
Kayvon Fatahalian[1] kayvonf@cs.stanford.edu

Abstract

Many real-world ML deployments face the challenge of training a rare category model with a small labeling budget. In these settings, there is often access to large amounts of unlabeled data, therefore it is attractive to consider semi-supervised or active learning approaches to reduce human labeling effort. However, prior approaches make two assumptions that do not often hold in practice; (a) one has access to a modest amount of labeled data to bootstrap learning and (b) every image belongs to a common category of interest. In this paper, we consider the scenario where we start with as-little-as five labeled positives of a rare category and a large amount of unlabeled data of which 99.9% of it is negatives. We propose an active semi-supervised method for building accurate models in this challenging setting. Our method leverages two key ideas: (a) Utilize human and machine effort where they are most effective; human labels are used to identify "needle-in-a-haystack" positives, while machine-generated pseudo-labels are used to identify negatives. (b) Adapt recently proposed representation learning techniques for handling extremely imbalanced human labeled data to iteratively train models with noisy machine labeled data. We compare our approach with prior active learning and semi-supervised approaches, demonstrating significant improvements in accuracy per unit labeling effort, particularly on a tight labeling budget.

1. Introduction

Training image classification models for a single (or small number of) rare category is common in real-world settings. For instance, autonomous vehicle development requires recognizing rare entities, like construction vehicles, in the video logs of a large fleet. A shopping application might need to recognize a particular type of apparel. Ecological monitoring requires recognition of rare animal species. A major challenge of building models for rare categories is acquiring training examples - precisely because they are rare! Fortunately, in many real-world scenarios one has access to a large amount of *unlabeled* data. Naively labeling the unlabeled data is unlikely to find many rare examples without significant human effort. Therefore, a natural strategy is to *interactively mine* the data by combining [24, 27, 16, 31, 40, 34] active [32] and semi-supervised [6, 46, 36, 43] learning techniques. However, prior active and semi-supervised learning approaches assume access to a modest amount of labeled data and assume that every image belongs to a common class of interest (Figure 1). In contrast, we are interested in building models for rare categories starting with only a few labeled positives (as little as 5) and where most (as much as 99.9%) of the unlabeled data is background. Simply put, our goal is to maximize model accuracy given a small fixed amount of human labels (500–1000).

Typical active and semi-supervised approaches use an initial labeled set to train a model for identifying and labeling relevant data. However, we have so few labeled positives for each rare category that it is difficult to train a deep model using only the initial labeled set. Prior work in the rare category setting resorts to training simple linear models using features from pre-trained deep networks [29, 11]. Unlike these approaches we show that it is feasible to improve deep features with a limited amount of labeled data using a combination of active and semi-supervised learning to address the following challenges:

- Because humans can only label a small fraction of the full dataset, we make use of semi-supervised learning to pseudo-labels additional examples. Pseudo-labeling positives works poorly because they are rare. **We let humans label "hard" examples that may contain positives and hard negatives and let machines pseudo-label "easy" negatives.**

[1]Stanford University [2]Carnegie Mellon University [3]Argo AI [4]Google Research

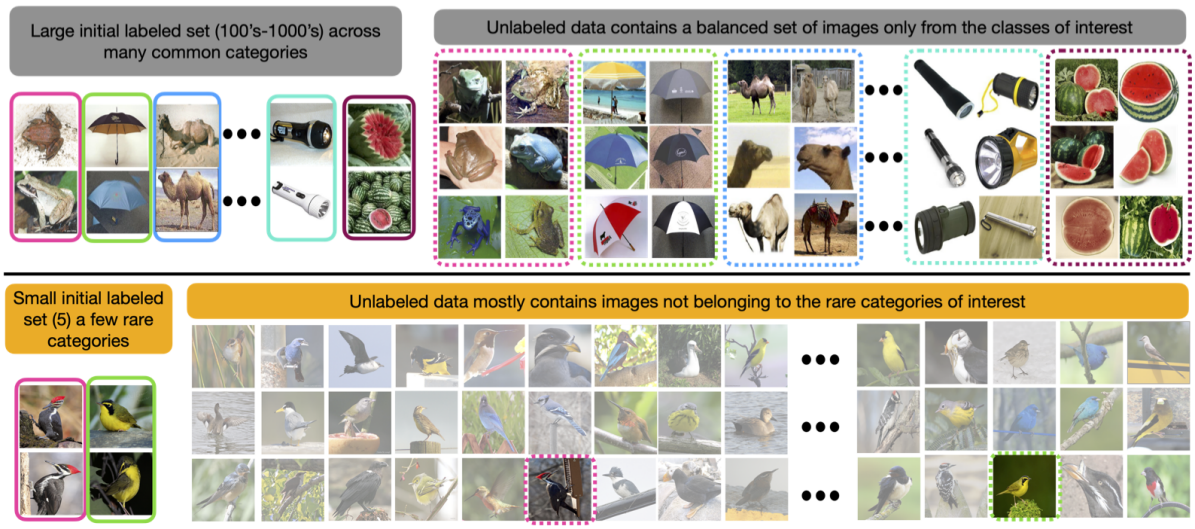


Figure 1: **Problem setup:** Most active and semi-supervised learning methods focus on balanced datasets where 100's-1000's of labeled examples are available to initialize learning (**top**). We argue that in practical ML deployment, one often wishes to learn a model for a rare category (where 99.9% of the data is background) from a small number of exemplars (**bottom**).

- Training on large amounts of pseudo-labeled negatives leads to difficult, highly-imbalanced learning. **We adapt background splitting [26], a recently proposed technique for learning from highly imbalanced human labeled datasets, for iterative semi-supervised learning with both human and machine labels.**

- Iteratively learning (deep) features significantly improves accuracy but is computationally expensive and adds significant delays between model updates and human labeling. **To reduce overall computational costs and latency between model updates and human labeling, we update the features at a low frequency but train linear models on cached features at a high frequency to pick the samples to query humans on.**

Figure 2 shows an overview of our approach, which synthesizes these key ideas into a human-in-loop system for building models for rare categories. Our system provides significant improvements in accuracy per unit labeling effort compared to prior approaches, particularly on a tight labeling budget.

2. Related Work

Our work overlaps research in *active learning*, *semi-supervised learning*, and the intersection of both of these methods with *deep learning*. Unlike prior work, we focus on rare categories where most of the available unlabeled data is background and where only a few positive exemplars are available to bootstrap the learning process.

Active learning The problem of choosing data to label to learn a model has be extensively studied in active learning [32] literature. Much of the early work has focused on training a classifier on top of pre-defined features where the unlabeled dataset has a *balanced* category distribution, and the goal is to build a model to classify all categories. State-of-the-art methods in this setting label data that the current classifier model is confused about (as measured by entropy or margin distance) [10] or data with the most information gain, often measured by the expected model gradient [32, 39]. When unlabeled data is balanced simple strategies like uniform random sampling are quite competitive [25]. Prior work has shown that uniform sampling as well as standard active learning techniques struggle in our imbalanced setting [3, 2]. Some work has explored active learning in the imbalanced/rare category setting [42, 4, 1], where unlabeled data only contains categories of interest and the imbalance studied is minor (2-99×) compared to the practical setting studied in this paper (384-10,000×).

Our work is most closely related to the Tropel system [29], which trains a rare-category classifier via active learning starting with just a few samples of the rare class. Tropel uses a fixed feature representation from a pre-trained deep model for classifier training. Similarly more recent work trains a logistic regression classifier on top of frozen deep features by limiting candidates for human labeling to the nearest-neighbors of the samples that have already been labeled [11]. In contrast, we update the deep feature representation used in the active process. Even though we have only a small amount of human labeled data, we make this update viable by also utilizing the unlabeled data.

Deep active learning & semi-supervised learning In traditional active learning methods, a classifier is learned on top of fixed features, whereas when training a complete deep model from actively-acquired samples the goal is to learn both a feature representation and a classifier. This difference requires fundamental changes to active learning techniques. Recent work [25] demonstrates that active learning with deep models in the common category setting benefits enormously from data augmentation or semi-supervised techniques using the unlabeled data. Therefore, straightforward adaptations of traditional active learning methods to training deep models [31, 17] are less efficient in their use of human samples than methods which augment deep active learning with *semi-supervised learning* [40, 34, 28, 13]. These combined techniques build on earlier work from the pre-deep-model era [24, 27, 16, 23].

Semi-supervised methods rely on an existing model to infer labels on the unlabeled data either using *proxy-labeling* or *graph-based methods* [28]. The existing model used to infer labels must be reasonably accurate for the methods to successfully improve the model. Most prior semi-supervised methods are evaluated in the many common category setting and start with 100's-1000's of labeled examples to train a good initial model [28, 20, 30]. However, with rare categories it is unrealistic to start with such a large number of labeled positive instances. In the realistic scenario that trains an initial model with just a few positive instances, we find that the model's *positive* predictions on unlabeled data are unreliable (the false positive rate is high). Therefore, we limit semi-supervised labeling to just a subset of the samples strongly predicted to be *negative*, confirming previous results in the classical setting [9]. The semi-supervised method that we use is a modified variant of the relatively simple *proxy-labeling* method [28].

Knowledge transfer and representation learning Techniques like knowledge distillation [7, 44, 30] enable transferring knowledge from a model in a related domain to learn better representations. We rely on a recent distillation based technique developed specifically for the rare category settings with extreme imbalance [26]. More recently self-supervised representation learning techniques have been successful for balanced datasets like ImageNet [5, 15, 7, 8]. We find that representations learned using such methods are good starting points but they need to be updated with supervision relevant to the rare categories of interest.

Few-shot learning Much work on active learning assumes a small but sizable portion of data is labeled (e.g., 5%), enough to *warm-start* with a reasonable model [33, 19]. In our setting, we assume only a small number of initial examples per class (5). This setting is close to that of a *cold start* in active learning [14, 21]. Our initial setup is similar to a few-shot learning problem, which often focuses on learning general representations [35] or aggressive data augmentation [41]. Rather than focusing on complex few-shot learning strategies our approach focuses on learning a good feature representations using unlabeled data echoing recent work [37].

Heterogeneous models for active learning Actively training deep models produces more accurate models but incurs a high computational cost. Prior work has shown that simpler models such as linear classifiers can be trained quickly compared to complex models such as deep CNNs, and are often good enough for sample selection in active learning [22, 12]. We use a similar approach and interleave linear classifier training with deep model training to retain advantages of both.

3. Approach

Our goal is to build an accurate model for a rare category given the following inputs: a small set of labeled positives (I_p); a large amount of unlabeled data (>99.9% of data is negative instances) from the target domain U; a pre-trained model M_{pre} from a related domain; and the ability to query a human for labels on a set of samples. Figure 2 provides an overview of our approach and Algorithm 1 gives concrete details. At a high level our approach iteratively builds a model by utilizing the best feature representation (F) and labeled positives (L_p), negatives (L_n), and pseudo negatives (W_n) available on hand to rapidly train linear models. Then we use the linear models to rank the unlabeled images and to query humans for labels on the top images in the ranking R. In addition to human labels, the linear model is used to accurately pseudo label a large number of *easy negatives* (W_n) by picking images from the bottom 50% of the ranking. Using fixed features not trained for the task limits the accuracy of the linear models. Therefore, the deep features are periodically updated at a slower rate by training the deep model (M^d) with both human and automatically labeled data. We use the recently proposed background splitting [26] technique to learn good representations even with the extreme imbalance encountered in the training data. The rest of the section elaborates on the motivation and the resulting design choices in the key components of our approach.

Bootstrapping the initial model and ranking. The first challenge we need to tackle is building a model for ranking the unlabeled data using the starting set of positives we call the prototype positives (I_p). Given only the prototype positives it is difficult to train a classifier since there are no negatives on hand. One could sample negatives randomly from the unlabeled data, but we find that randomly sampled negatives are not effective for building an accurate classi-

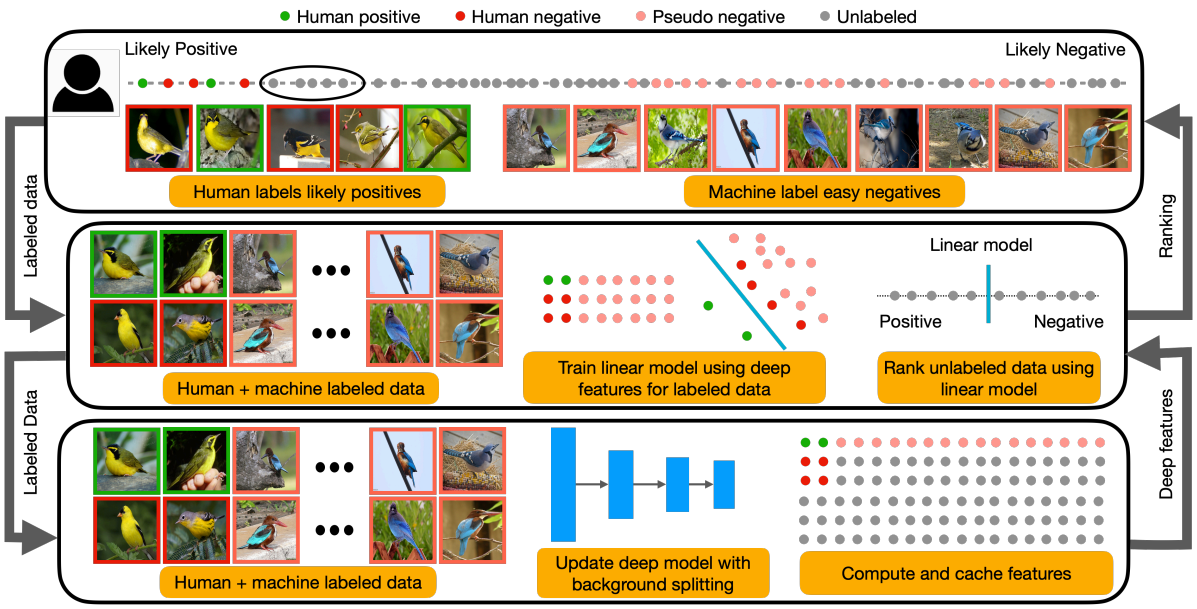


Figure 2: **Overview:** Our approach builds rare category models by alternating (top cycle) between selecting unlabeled samples for a human to label and training a linear model on top of frozen deep features from labeled images. Our pipeline focuses human effort on images likely to contain positives by ranking unlabeled images with the linear model and querying humans on high scoring unlabeled images. In addition to human labels, the linear model is used to automatically label a large number of *easy negatives* by picking images from the bottom 50% of the ranking. Using fixed features not trained for the task limits the accuracy of the linear models. Therefore, the deep feature representation is periodically updated at a slower rate (bottom cycle) by training the deep model with both human and automatically labeled data.

fier. Instead it is more effective to rely on feature similarity to the positive instances for finding hard negative examples. Therefore, we generate a feature-similarity based ranking using the prototype positives. **`cacheFeatures`** (Line 1) uses the pre-trained model M^d_{pre} to pre-compute the features F for all the samples in the unlabeled data U. Then **`computeFeatureCentroid`** (Line 3) normalizes the feature representations of all the prototype positive examples I_p and computes an average feature representation c (category prototype) for the rare category. Finally, **`rankByCosineSimilarity`** (Line 4) generates the initial ranking R_{pos} of unlabeled images in U by sorting them by cosine similarity to the category prototype c.

Querying for human labels on high-ranking samples. Given the initial bootstrapped model c or the linear models (M^l) in later iterations we need to choose which unlabeled images to present to the human for labeling. The initial ranking (Line 4) is based on similarity and surfaces data that is visually similar to the category of interest. However, the ranking can be unreliable (especially for fine-grained categories) since the similarity is computed using only a few positive instances (I_p) based on a representation learned for a different task (ImageNet classification or a self-supervised auxiliary task). Even in subsequent iterations we only request a small number of ($B = 10$) additional labels every iteration to update the ranking model (M^l). For instance, Figure 2 top shows the highest ranking unlabeled images during the active loop. Most of the top images surfaced by the model are visually similar, but only a small fraction of those images belongs to the category of interest making it difficult to automatically label positives. Therefore, we focus human labeling effort on the top scoring positive images in the ranking.

Each iteration of the inner loop j, **`queryHumanTopK`** (Lines 10,12) queries the human for labels on the B top ranked images, where B is the labeling budget per iteration. The human labeled positives and negatives H_p, H_n are added to the appropriate labeled sets L_p, L_n and removed from the unlabeled data pool. As we request more labels in the iterative process and update the model, the model becomes more accurate and scores easy positives highly. Once the model is reasonably accurate, labeling more easy positives doesn't improve the model much, but the model is then able to identify samples near the true margin (hard positives and hard negatives). Based on this intuition, we use a simple heuristic to estimate model quality and adaptively choose a strategy for picking the samples for human labeling. The key insight is that a high quality model finds more positives than negatives. Our approach keeps track of the number

Algorithm 1: Our rare category active approach.

Input: U, I_p, M^d_{pre}, B, N, Q, f_a
Output: $M^d_{1...N}$

1 $F \leftarrow$ **cacheFeatures**(M^d_{pre}, U)
2 $A_l \leftarrow$ **auxiliaryLabels**(M^d_{pre}, U)
3 $c \leftarrow$ **computeFeatureCentroid**(M^d_{pre}, I_p)
4 $R_{pos} \leftarrow$ **rankByCosineSimilarity**(c, F)
5 $L_p \leftarrow I_p$, $L_n \leftarrow \{\}$, $W_n \leftarrow \{\}$
6 $M^d_0 \leftarrow M^d_{pre}$
7 **for** $i \leftarrow 1$ **to** N **do**
8 **for** $j \leftarrow 1$ **to** Q **do**
9 **if** $|L_p| < |L_n|$ **then**
10 $H_p, H_n \leftarrow$ **queryHumanTopK**(B, R_{pos})
11 **else**
12 $H_p, H_n \leftarrow$ **queryHumanTopK**(B, R_{ent})
13 $L_p \leftarrow H_p \cup L_p$, $L_n \leftarrow H_n \cup L_n$
14 $W_n \leftarrow$ **pseudoNegativeLabels**(R_{pos}, f_a)
15 $M^l \leftarrow$ **trainLinearModel**(F,L_p,L_n,W_n)
16 $U \leftarrow U - (H_p \cup H_n)$
17 $R_{pos} \leftarrow$ **sortPositiveScore**(M^l, U)
18 $R_{ent} \leftarrow$ **sortMarginDistance**(M^l, U)
19 $M^d_i \leftarrow$ **trainBGSplit**(M^d_{i-1},L_p, L_n, W_n, A_l)
20 $F \leftarrow$ **cacheFeatures**(M^d_i, U)

of positives and negatives labeled so far ($|L_p|, |L_n|$). If the number of positives labeled so far is less than the number of negatives, we ask the human to label the most-likely-positive images (R_{pos}), otherwise we ask the human to label the images closest to the linear model's margin (R_{ent}).

Generating negative pseudo labels on low ranking samples. Given only the small number of human labeled images ($|L_p|, |L_n|$) it is difficult to improve the deep feature representation. Our approach addresses this issue by machine labeling a large subset of the unlabeled images U. Firstly, our approach avoids labeling errors by restricting pseudo labels to negatives and more precisely the low ranked samples in the current ranking R_{pos} which are unlikely to be positives. Secondly, our algorithm takes a progressive approach to machine labeling based on the number of human-labeled positives and negatives (L_p and L_n). The routine **pseudoNegativeLabels** (Line 14) chooses $f_a(|L_p| + 0.1|L_n|)$ samples uniformly from the bottom half of the ranking as pseudo negatives. f_a is a scalar hyper parameter that controls the amount of pseudo labels relative to the human-labeled data the algorithm uses. The goal of this heuristic is to keep the label errors relative to the amount of human-labeled data low while maintaining diversity in the negatives. Although easy negatives can be automatically labeled reliably with simple heuristics, similar heuristics cannot be used to generate automatic labels for positives of the rare category. We show that even conservatively pseudo labeling positives results in lower accuracy in the active process compared to our approach.

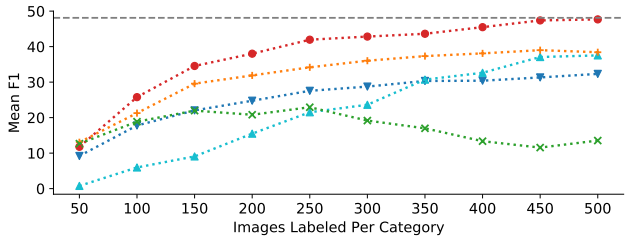
Updating feature representation with extreme imbalance. Training the deep models M^d_i with a small number of human labeled images and a large number of pseudo labeled negatives remains challenging due to the extreme imbalance in the training data. Our approach mitigates this issue while distilling information from a related domain by using a recent deep-model training technique called background splitting [26] designed for coping with extreme imbalance. **trainBGSplit** (Line 19) trains the deep representation using all the human-labeled and pseudo-labeled data, plus auxiliary labels A_l computed using the pre-trained model M^d_{pre}.

Interleaved feature representation and ranking updates. In principle, we can update the deep model with every human label to make best use of subsequent human effort. However, updating the deep representation frequently is computationally expensive and gives diminishing returns as we show in the supplemental. Conversely, training linear models on fixed features is computationally cheap but results in lower quality models. Our approach strikes a balance between the two extremes by interleaving low frequency updates of the feature representation (every QB human labels) with high frequency training of a linear model (every B human labels) used to update the ranking. The outer loop i corresponds to feature updates and the inner loop j corresponds to ranking updates with a fixed feature representation. The parameters N and Q specify the number of feature and ranking update iterations respectively. The routine **trainBGSplit** starts with the model from the previous iteration and only trains it for a few epochs on the current labeled data (L_p, L_n, W_n). Continuous training of the deep model across the active iterations keeps the training cost close to that of training a fully supervised deep model just once on all the data. After every update the features for the unlabeled data are cached (Line 20) for training linear models used to choose images for human labeling. In between the feature updates, the routine **trainLinearModel** uses the current features to rapidly train a linear model (M^l) using both human and pseudo negative labels. The resulting linear model is used to update the ranking used to pick images for human labeling as well as pseudo negatives.

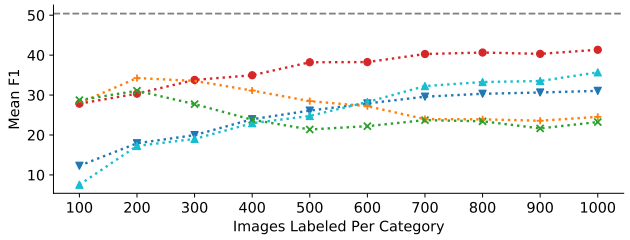
4. Evaluation

Our approach has three components: active sampling, pseudo labeling and deep model training on imbalanced data. Prior work has largely focused on individual compo-

Approach		Pseudo Positives	Pseudo Negatives	BG Splitting	Train Features	Active Learning
Ours	○	✗	✓	✓	✓	✓(adaptive)
KD-Semi	+	✓	✓	✓	✓	✓(adaptive)
DeepProp	×	✓	✓	✓	✓	✓(adaptive)
Tropel-Deep	△	✗	✗	✗	✓	✓(most-likely-positive)
Tropel	▽	✗	✗	✗	✗	✓(most-likely-positive)
Fully supervised		✗	✗	✓	✓	✗



(a) iNaturalist 50 categories (each is ∼0.01-0.26% of data)



(b) Places 20 categories (each is ∼0.19-0.27% of data)

Figure 3: **Comparison to baselines.** Plots show average F1 accuracy of our approach and baselines on 50 iNaturalist and 20 Places categories as a function of human labeling effort. **Our** approach gives higher accuracy per human effort compared to baselines, especially when a small number of images are labeled. The **Tropel** baseline trains a linear model on fixed features and is competitive with **Tropel-Deep** baseline in the low data regime. **DeepProp** and **KD-Semi** baselines pseudo label positives in addition to negatives. The false positives in the pseudo positive labels leads to lower accuracy compared to our approach.

nents of our approach. We compare our approach to prior work by replacing individual components with prior methods and evaluating the overall performance. We also present ablation studies for key ideas.

4.1. Experimental setup

We evaluate our approach for building rare category models using two datasets: the iNaturalist [38] fine-grained species recognition dataset and the Places [45] scene classification dataset. We randomly choose 50 categories from the iNaturalist dataset and 20 categories from the Places dataset for evaluation as rare categories. For the iNaturalist dataset we restrict the random choice to categories with more than 50 instances in the training set, so that the active learning process has something to find. Given this setup, positive instances for individual categories are extremely rare in both datasets. For iNaturalist, the number of positives for each category is in the range 50-1500 (∼0.01-0.26% of the training set). For Places, the categories are more uniformly distributed, with ∼3500-5000 instances per category (∼0.19-0.27% of the training set). For each category we pick five randomly chosen positives as the initial labeled set and treat the rest of the training data as the unlabeled pool of data. We evaluate the trained models on the full validation set of each dataset. All images not in a category of interest are considered negatives in our evaluation i.e., our validation set reflects the imbalance in the training set. We use the F1 metric for evaluation instead of classification accuracy since classification accuracy is easily gamed for rare categories by just predicting everything as negative.

Rare-category active learning baselines: **Tropel** [29] is a prior active learning approach for rare categories. **Tropel** iteratively trains a linear model (on top of frozen pre-trained deep features) by labeling the top-k scoring predictions from the current linear model. We also evaluate a new variant, **Tropel-Deep**, which trains a deep model instead of a linear model at each iteration. **Tropel-Deep** uses the same incremental deep-model training that we use in our approach.

Semi-supervised baselines As the results show, our method outperforms **Tropel** (because we train a deep model) and **Tropel-Deep** (because our method augments the human labels with pseudo-labels). This second result raises the question as to whether traditional semi-supervised techniques could perform just as well as our enhanced methods. To study this question, we implemented additional baselines based on label propagation [20, 34] and knowledge distillation [7, 44, 30]. We replace our pseudo negative labeling approach with these approaches for generating automatic labels, naming them **DeepProp** and **KD-Semi**. For the DeepProp baseline, instead of only using the pseudo negatives W_n to train the deep model, we also use pseudo positive labels generated with label propagation [20]. For the KD-Semi baseline, instead of only using the pseudo negatives W_n produced by the linear model, we use the high confidence predictions of the deep model from the previous iteration on the unlabeled data as pseudo positive and negative labels. More details on the baselines are provided in the supplemental.

Implementation and configuration details We use a ResNet-50 model [18] pre-trained on ImageNet or using self-supervised methods (M^d_{pre}) to initialize the deep model and compute feature representations for all the unlabeled

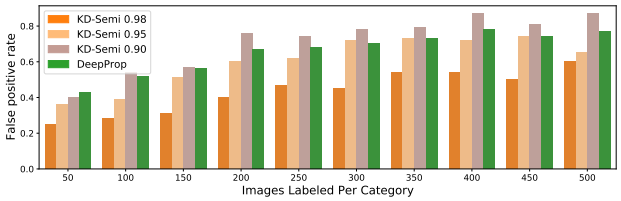


Figure 4: **Analysis of automatic positives.** Plot shows the average false positive rate in the automatic positives labeled by **KD-Semi** and **DeepProp** baselines for the 50 iNaturalist categories. Lowering the confidence threshold for automatic positives with **KD-Semi** increases the false positive rate. The high false positive rates showcase the difficulty in automatically labeling positives.

data. The same ResNet-50 model is used to generate labels for the background splitting auxiliary loss. We run our active approach and its variants with $N = 10$, $Q = 5$, and $B = 10$ for iNaturalist and $N = 10$, $Q = 5$, and $B = 20$ for Places unless specified otherwise. We use the same set of initial positives (I_p) for each category across all the methods and set f_a to 100 unless specified otherwise. In the supplemental we show that our approach is robust to a range of values for f_a.

Ideally, we would evaluate our approach and all the baselines individually for each category. However, due to computational limitations it is not feasible to train deep models with pseudo negatives individually for each category. Therefore, we run our approach and baselines for groups of ten categories at a time to reduce the compute costs. We maintain the same grouping of categories across all the experiments. When training the deep model we use binary cross entropy loss for each category as opposed to cross entropy loss across all the ten categories and supervise the model with only binary labels for each category. The linear models used to rank unlabeled data are trained **independently for each category** and humans are queried for binary labels **for each category separately**. Therefore, images might only be labeled for one of the ten categories. When training the deep models we deal with the missing labels by masking the corresponding binary loss for categories that are not labeled.

When updating the feature representation we train the deep model for 15 epochs on the currently labeled data in the active loop. In the $N = 10$, $Q = 5$, and $B = 10$ configuration we thus run 150 total epochs of training. However, on average an epoch in our approach only uses a quarter of the unlabeled data. Therefore, the overall cost of training is approximately equal to 40 epochs of training on the full dataset. For the fully supervised baseline, we train for 50 epochs. More details on the training schedules are provided in the supplemental.

4.2. Comparison with baselines

We compare our approach with prior work by measuring model accuracy of different methods as a function of human labeling effort. Figure 3 shows mean F1 accuracy of our approach and several baselines. The accuracy is computed on 50 and 20 categories of interest in the iNaturalist and Places datasets. We measure labeling effort as the number of binary labels per category. Our approach gives higher accuracy for the same human labeling effort compared to baselines, especially when a small number of images are labeled. The **Tropel** baseline trains a linear model on fixed features computed using a deep model pre-trained on ImageNet. The **Tropel-Deep** baseline trains a deep model only using the human labeled data. When only a small amount of human labeled data is available training a deep model can overfit leading to low accuracy. As one can see, **Tropel-Deep** performs worse than **Tropel** when a small amount of images are labeled. In contrast, our approach uses pseudo negatives in addition to human labeled data allowing us to update the feature representation without overfitting. On the iNaturalist categories, our approach matches fully supervised accuracy when 500 images per category are labeled since the dataset contains less than 500 positives for each of the categories. On the Places categories, there is a significant gap with fully supervised accuracy but this expected since there are 5000 positives per category and we only label a total of 1000 images per category.

DeepProp and **KD-Semi** use pseudo labels on the unlabeled data to update the deep feature representation. Unlike our approach these baselines do not restrict pseudo labels to easy negatives. **DeepProp** propagates labels to all the unlabeled data based on the feature similarity graph and assigns weights to the pseudo labels, whereas **KD-Semi** uses the current model to predict pseudo labels on the unlabeled data. Both **DeepProp** and **KD-Semi** perform worse that our approach due to errors in the labels propagated on positives. Figure 4 shows the false positive rate of **KD-Semi** and **DeepProp** on the iNaturalist categories. A high false positive rate is expected since the models are trained from very few labeled examples. The false positive rate keeps increasing as errors in labeling propagate and distort the definition of the category. **KD-Semi** can be tuned to reduce errors in the positive pseudo labels by increasing the confidence threshold at which the labels are trusted. Figure 4 shows the false positive rates for **KD-Semi** at different confidence thresholds. Even with confidence threshold as high as 0.98 the false positive rate is significant and is even worse with lower confidence thresholds. Setting the confidence threshold higher than 0.98 effectively results in almost no pseudo positive labels and tends to our strategy in the limit. **DeepProp** performs worse than **Tropel** due to the high amount of errors in label propagation which is difficult to control unlike **KD-Semi** as shown in Figure 4.

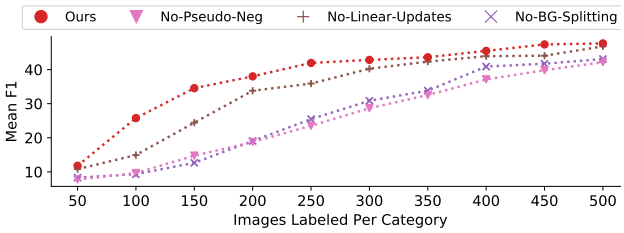


Figure 5: **Ablation study.** Plot shows average F1 accuracy for variants of our approach on 50 iNaturalist categories as a function of human labeling effort. Omitting pseudo negatives **No-Pseudo-Neg** or background splitting to handle imbalance **No-BG-Splitting** results in lower model accuracy, especially when a small amount of images are labeled. Only updating deep models every 50 labeled images **No-Linear-Updates** also results in lower accuracy.

4.3. Ablation study

We created variants of our approach to evaluate the impact of the different components. Figure 5 shows F1 accuracy for variants of our approach on 50 categories from the iNaturalist dataset as a function of human labeling effort. **No-Pseudo-Neg** and **No-BG-Splitting** are variants without pseudo negatives and background splitting respectively. As one can see omitting the pseudo negatives when updating the deep representation results in lower accuracy for the same human labeling effort. As one would expect, the effect of pseudo negatives and background splitting is more pronounced when a small amount of images are labeled. Since our approach only uses a fraction of the easy negatives along with the human labeled positives and hard negatives for training, it also reduces the imbalance between positive and negative instances when compared to fully supervised training using all the data. **No-Linear-Updates** only updates the linear model once every representation update; this corresponds to the parameter setting $N = 10, B = 50, Q = 1$. Skipping the linear model updates results in a lower accuracy for the same labeling effort. The linear model updates enable quickly adapting the ranking used to choose images for human labeling and hence lead to better overall model.

4.4. Using self-supervised representations

Until now we have evaluated our approach using a model pre-trained on ImageNet using supervised labels shown as **Ours-Sup-ImageNet** in Figure 6. Instead **Ours-Self-iNat** and **Ours-Self-ImageNet** use self-supervised models trained on iNaturalist and ImageNet with SwAV [5] as the pre-trained model M^d_{pre}. We generate auxiliary labels for background splitting by clustering the iNaturalist data with the self-supervised feature representations. Similarly, **Tropel-Self-iNat** and **Tropel-Self-ImageNet** actively train linear models using fixed self-supervised representations.

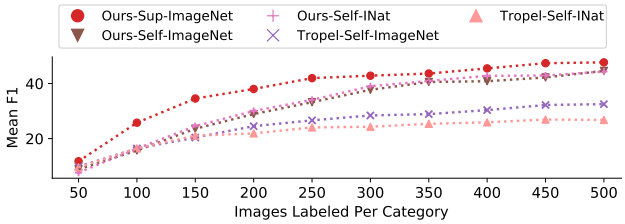


Figure 6: **Self-supervised representations.** Plot shows average F1 accuracy of our approach and Tropel on 50 iNaturalist categories as a function of human labeling effort. **Ours-Self-iNat** and **Ours-Self-ImageNet** use self-supervised models trained on iNaturalist and ImageNet as the starting point. **Tropel-Self-iNat** and **Tropel-Self-ImageNet** train linear models using self-supervised representations. Our approach is more label efficient and shows the importance of actively updating features. **Ours-Sup-ImageNet** uses ImageNet supervised pre-training as the starting point and outperforms self-supervised baselines.

Our approach produces more accurate models showing the importance of updating feature representation with actively labeled data even when starting with a self-supervised representation trained on the same data. These results show that our approach when combined with self-supervised pre-training produces accurate models with just 500 binary labels for each category. However, our approach produces better models with supervised pre-trained representations on ImageNet. This suggests room for improvement in self-supervised representation learning approaches.

5. Conclusion

In many real world scenarios it is often necessary to build a model for a rare category starting from a small positive set with access to a large unlabeled data collection which contain instances of the rare category. In this setting, we show that leveraging a combination human labels for positives and hard negatives and machine labels for easy negatives is crucial for building accurate models under a tight labeling budget. We find that techniques that work well in the common multi category setting like propagating positive labels or training a deep model using standard losses do not work off-the-shelf and need to be adapted to this real world setting. We hope that our work showcases the challenges presented by rare categories in real-world settings and encourages future work.

Acknowledgments This work is supported by the National Science Foundation (NSF) under III-1908727 and CCF-1937301. Deva Ramanan and Ravi Teja Mullapudi were supported by the CMU Argo AI Center for Autonomous Vehicle Research. We thank Florian Schroff and Hartwig Adam for their support of the work and Afshin Rostamizadeh and Giulia DeSalvo for their advice on active learning baselines.

References

[1] Umang Aggarwal, Adrian Popescu, and Céline Hudelot. Active learning for imbalanced datasets. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 1428–1437, 2020. 2

[2] Josh Attenberg and Seyda Ertekin. Class imbalance and active learning. *Imbalanced Learning: Foundations, Algorithms, and Applications*, pages 101–149, 2013. 2

[3] Josh Attenberg and Foster Provost. Why label when you can search? alternatives to active learning for applying human resources to build classification models under extreme class imbalance. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 423–432, 2010. 2

[4] Mausam C Lin. Active learning with unbalanced classes & example-generated queries. In *AAAI Conference on Human Computation*, 2018. 2

[5] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *NeurIPS*, 2020. 3, 8

[6] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. *Semi-Supervised Learning*. MIT Press, 2006. 1

[7] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020. 3, 6

[8] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 3

[9] Yukun Chen and Subramani Mani. Active learning for unbalanced data in the challenge with multiple models and biasing. In *Active Learning and Experimental Design workshop In conjunction with AISTATS 2010*, pages 113–126, 2011. 3

[10] Galen Chuang, Giulia DeSalvo, Lazaros Karydas, Jean-François Kagy, Afshin Rostamizadeh, and A Theeraphol. Active learning empirical study. NeurIPS 2019 Workshop on Learning with Rich Experience: Integration of Learning Paradigms, 2019. Available at workshop website http://sites.google.com/view/neurips2019lire. 2

[11] Cody Coleman, Edward Chou, Sean Culatana, Peter Bailis, Alexander C Berg, Roshan Sumbaly, Matei Zaharia, and I Zeki Yalniz. Similarity search for efficient active learning and search of rare concepts. *arXiv preprint arXiv:2007.00077*, 2020. 1, 2

[12] Cody Coleman, Christopher Yeh, Stephen Mussmann, Baharan Mirzasoleiman, Peter Bailis, Percy Liang, Jure Leskovec, and Matei Zaharia. Selection via proxy: Efficient data selection for deep learning. *ICLR*, 2020. 3

[13] Mingfei Gao, Zizhao Zhang, Guo Yu, Sercan Ö Arık, Larry S Davis, and Tomas Pfister. Consistency-based semi-supervised active learning: Towards minimizing labeling cost. In *European Conference on Computer Vision*, pages 510–526. Springer, 2020. 3

[14] Roman Garnett, Yamuna Krishnamurthy, Xuehan Xiong, Jeff Schneider, and Richard Mann. Bayesian optimal active search and surveying. *arXiv preprint arXiv:1206.6406*, 2012. 3

[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. *arXiv preprint arXiv:2006.07733*, 2020. 3

[16] Mohamed Farouk Abdel Hady and Friedhelm Schwenker. Combining committee-based semi-supervised and active learning and its application to handwritten digits recognition. In *International Workshop on Multiple Classifier Systems*, pages 225–234. Springer, 2010. 1, 3

[17] Elmar Haussmann, Michele Fenzi, Kashyap Chitta, Jan Ivanecky, Hanson Xu, Donna Roy, Akshita Mittel, Nicolas Koumchatzky, Clement Farabet, and Jose M Alvarez. Scalable active learning for object detection. *arXiv preprint arXiv:2004.04699*, 2020. 3

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6

[19] Peiyun Hu, Zachary C Lipton, Anima Anandkumar, and Deva Ramanan. Active learning with partial feedback. *arXiv preprint arXiv:1802.07427*, 2018. 3

[20] Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Label propagation for deep semi-supervised learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5070–5079, 2019. 3, 6

[21] Shali Jiang, Roman Garnett, and Benjamin Moseley. Cost effective active search. In *Advances in Neural Information Processing Systems*, pages 4880–4889, 2019. 3

[22] David D Lewis and Jason Catlett. Heterogeneous uncertainty sampling for supervised learning. In *Machine learning proceedings 1994*, pages 148–156. Elsevier, 1994. 3

[23] Hui Li, Xuejun Liao, and Lawrence Carin. Active learning for semi-supervised multi-task learning. In *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 1637–1640. IEEE, 2009. 3

[24] Andrew Kachites McCallumzy and Kamal Nigamy. Employing em and pool-based active learning for text classification. In *Proc. International Conference on Machine Learning (ICML)*, pages 359–367. Citeseer, 1998. 1, 3

[25] Sudhanshu Mittal, Maxim Tatarchenko, Özgün Çiçek, and Thomas Brox. Parting with illusions about deep active learning. *arXiv preprint arXiv:1912.05361*, 2019. 2, 3

[26] Ravi Teja Mullapudi, Fait Poms, William R. Mark, Deva Ramanan, and Kayvon Fatahalian. Background splitting: Finding rare classes in a sea of background, 2020. 2, 3, 5

[27] Ion Muslea, Steven Minton, and Craig A Knoblock. Active+ semi-supervised learning= robust multi-view learning. In *ICML*, volume 2, pages 435–442, 2002. 1, 3

[28] Yassine Ouali, Céline Hudelot, and Myriam Tami. An overview of deep semi-supervised learning. *arXiv preprint arXiv:2006.05278*, 2020. 3

[29] Genevieve Patterson, Grant Van Horn, Serge Belongie, Pietro Perona, and James Hays. Tropel: Crowdsourcing de-

tectors with minimal training. In *Third AAAI Conference on Human Computation and Crowdsourcing*, 2015. 1, 2, 6

[30] Sylvestre-Alvise Rebuffi, Sebastien Ehrhardt, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Semi-supervised learning with scarce annotations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 762–763, 2020. 3, 6

[31] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017. 1, 3

[32] Burr Settles. Active learning literature survey (computer sciences technical report 1648). *University of Wisconsin-Madison*, 2010. 1, 2

[33] Yanyao Shen, Hyokun Yun, Zachary C Lipton, Yakov Kronrod, and Animashree Anandkumar. Deep active learning for named entity recognition. *arXiv preprint arXiv:1707.05928*, 2017. 3

[34] Oriane Siméoni, Mateusz Budnik, Yannis Avrithis, and Guillaume Gravier. Rethinking deep active learning: Using unlabeled data at model training. *arXiv preprint arXiv:1911.08177*, 2019. 1, 3, 6

[35] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in neural information processing systems*, pages 4077–4087, 2017. 3

[36] 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. *arXiv preprint arXiv:2001.07685*, 2020. 1

[37] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: a good embedding is all you need? *arXiv preprint arXiv:2003.11539*, 2020. 3

[38] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8769–8778, 2018. 6

[39] Carl Vondrick and Deva Ramanan. Video annotation and tracking with active learning. In *Advances in Neural Information Processing Systems*, pages 28–36, 2011. 2

[40] Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. Cost-effective active learning for deep image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(12):2591–2600, 2016. 1, 3

[41] Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7278–7286, 2018. 3

[42] Manfred K Warmuth, Jun Liao, Gunnar Rätsch, Michael Mathieson, Santosh Putta, and Christian Lemmen. Active learning with support vector machines in the drug discovery process. *Journal of chemical information and computer sciences*, 43(2):667–673, 2003. 2

[43] Colin Wei, Kendrick Shen, Yining Chen, and Tengyu Ma. Theoretical analysis of self-training with deep networks on unlabeled data. *arXiv preprint arXiv:2010.03622*, 2020. 1

[44] 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. 3, 6

[45] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 6

[46] Xiaojin Zhu and Andrew B Goldberg. Introduction to semi-supervised learning (synthesis lectures on artificial intelligence and machine learning). *Morgan and Claypool Publishers*, 14, 2009. 1