



# Exploring the Limits of Weakly Supervised Pretraining

Dhruv Mahajan, Ross Girshick<sup>(✉)</sup>, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten

Facebook, Menlo Park, USA  
rbg@fb.com

**Abstract.** State-of-the-art visual perception models for a wide range of tasks rely on supervised pretraining. ImageNet classification is the de facto pretraining task for these models. Yet, ImageNet is now nearly ten years old and is by modern standards “small”. Even so, relatively little is known about the behavior of pretraining with datasets that are multiple orders of magnitude larger. The reasons are obvious: such datasets are difficult to collect and annotate. In this paper, we present a unique study of transfer learning with large convolutional networks trained to predict hashtags on *billions* of social media images. Our experiments demonstrate that training for large-scale hashtag prediction leads to excellent results. We show improvements on several image classification and object detection tasks, and report the highest ImageNet-1k single-crop, top-1 accuracy to date: 85.4% (97.6% top-5). We also perform extensive experiments that provide novel empirical data on the relationship between large-scale pretraining and transfer learning performance.

## 1 Introduction

Nearly all state-of-the-art visual perception algorithms rely on the same formula: (1) pretrain a convolutional network on a large, manually annotated image classification dataset and (2) finetune the network on a smaller, task-specific dataset. This formula [1–3] has been in wide use for several years and led to impressive improvements on numerous tasks. Examples include: object detection [1, 4], semantic segmentation [5, 6], human pose estimation [7, 8], video recognition [9], monocular depth estimation [10], and so on. In fact, it is so effective that it would now be considered foolhardy *not* to use supervised pretraining.

The ImageNet dataset [11] is the de facto pretraining dataset. While there are studies analyzing the effects of various ImageNet pretraining factors on transfer learning (e.g., [12, 13]) or the use of different datasets that are of the same size magnitude as ImageNet (e.g., [14, 15]), relatively little is known about pretraining on datasets that are *multiple* orders of magnitude larger ([16, 17] are the largest

**Electronic supplementary material** The online version of this chapter ([https://doi.org/10.1007/978-3-030-01216-8\\_12](https://doi.org/10.1007/978-3-030-01216-8_12)) contains supplementary material, which is available to authorized users.

studies to date). The reasons for this are numerous: few such datasets exist, building new datasets is labor intensive, and large computational resources are needed to conduct experiments. Yet, given the central role of pretraining it is important to expand our scientific knowledge in this domain.

This paper tries to address this complex issue by studying an unexplored data regime: *billions of images “labeled” in the wild with social media hashtags*. This data source has the advantage of being large and continuously growing, as well as “free”, from an annotation perspective, because no manual labeling is required. However, the data source also has potential disadvantages: hashtags may be too noisy to serve as an effective supervisory signal and the image distribution might be biased in ways that harm transfer learning. It is not a priori obvious that training on this data will yield good transfer learning results.

The main result of this paper is that without manual dataset curation or sophisticated data cleaning, models trained on billions of Instagram images using thousands of distinct hashtags as labels exhibit excellent transfer learning performance. For example, we observe improvements over the state-of-the-art for image classification and object detection, where we obtain a single-crop, top-1 accuracy of 85.4% on the ImageNet-1k image-classification dataset and 45.2% AP on the COCO object-detection dataset [18], compared to 79.8% and 43.7%, respectively, when training (or pretraining) the same models on ImageNet-1k. Our primary goal, however, is to contribute novel experimental data about this previously unexplored regime. To that end, we conduct numerous experiments that reveal interesting trends. For example, we find that “hashtag engineering” (i.e., collecting images tagged with a specific subset of hashtags) is a promising new direction for improving transfer learning results, that training on large-scale hashtag data is unexpectedly robust to label noise, and that the features learned allow a simple linear classifier to achieve state-of-the-art ImageNet-1k top-1 accuracy of 83.6% without any finetuning (compared to 84.2% with finetuning).

## 2 Scaling up Supervised Pretraining

In our experiments, we train standard convolutional network architectures to predict hashtags on up to 3.5 billion public Instagram images. To make training at this scale practical, we adopt a distributed synchronous implementation of stochastic gradient descent with large (8k image) minibatches, following Goyal et al. [19]. We experiment on a variety of datasets, which we describe next.

### 2.1 Instagram Datasets

We use a simple data collection pipeline: (1) We select a set of hashtags. (2) We download images that are tagged with at least one of these hashtags. (3) Then, because multiple hashtags may refer to the same underlying concept, we apply a simple process that utilizes WordNet [20] synsets to merge some hashtags into a single canonical form (e.g., #brownbear and #ursusarctos are merged). (4)

Finally, for each downloaded image, we replace each hashtag with its canonical form and discard any hashtags that were not in the selected set. The canonical hashtags are used as labels for training and evaluation.

By varying the selected hashtags and the number of images to sample, we can construct a variety of datasets of different sizes and visual distributions. Table 1 summarizes the datasets used in our experiments. Each dataset is named by completing a template, *role-source-I-L*, that indicates its role (training, validation, testing), source (IG for Instagram, IN for ImageNet, etc.), number of images  $I$ , and number of labels  $L$ . We use approximate image and label counts for convenience, for example “train-IG-940M-1.5k” is an Instagram dataset for training with  $\sim 940\text{e}6$  images and  $\sim 1,500$  labels. We omit the role and image count when it is clear from context or not useful to present.

We design three hashtag sets for the Instagram data: (1) A  $\sim 1.5\text{k}$  set with hashtags from the standard 1,000 IN-1k synsets (each synset contains at least one synonym, hence there are more hashtags than synsets). (2) A  $\sim 17\text{k}$  set with hashtags that are synonyms in any of the noun synsets in WordNet. And (3) an  $\sim 8.5\text{k}$  set with the most frequent hashtags from the 17k set. The hashtag set sizes are measured after merging the hashtags into their canonical forms. We hypothesize that the first set has a visual distribution similar to IN-1k, while the other two represent more general visual distributions covering fine-grained visual categories. Details of how these hashtags are selected and how the merging process works are given in supplemental material.

**Table 1. Summary of image classification datasets.** Each dataset is named with a template, *role-source-I-L*, that indicates its role (training, validation, testing), source, number of images  $I$ , and number of labels  $L$ .

Name template	Description
train-IG- $I$ -1.5k	Instagram training set of $I$ images and $\sim 1.5\text{k}$ hashtags from ImageNet-1k
train-IG- $I$ -8.5k	Instagram training set of $I$ images and $\sim 8.5\text{k}$ hashtags from WordNet
train-IG- $I$ -17k	Instagram training set of $I$ images and $\sim 17\text{k}$ hashtags from WordNet
train-IN-1M-1k	The standard ImageNet-1k ILSVRC training set with 1.28M images
val-IN-50k-1k	The standard ImageNet-1k ILSVRC validation set with 50k images
train-IN- $I$ - $L$	Extended ImageNet training set of $I$ images and $L \in \{5\text{k}, 9\text{k}\}$ labels
val-IN- $I$ - $L$	Extended ImageNet validation set of $I$ images and $L \in \{5\text{k}, 9\text{k}\}$ labels
train-CUB-6k-200	The Caltech-UCSD Birds-200-2011 training set
val-CUB-6k-200	The Caltech-UCSD Birds-200-2011 validation set
train-Places-1.8M-365	The Places365-Standard training set (high-resolution version)
val-Places-37k-365	The Places365-Standard validation set (high-resolution version)
train-COCO-115k-80	The standard COCO detection training set (2017 version)
val-COCO-5k-80	The standard COCO detection validation set (2017 version)
test-COCO-20k-80	The standard COCO detection test-dev set (2017 version)

**Image Deduplication.** When performing transfer learning, it is essential to understand and properly address overlap between training and test sets. Overlap

can exist because images may come from the same underlying sources (e.g., Wikipedia, Flickr, Google). For instance,  $\sim 5\%$  of the images in the val-CUB-6k-200 set [21] also appear in train-IN-1M-1k, and 1.78% of images in val-IN-50k-1k set are in the JFT-300M training set [17]. To address this issue, we performed the following deduplication procedure: we compute R-MAC features [22, 23] for all candidate images using a ResNet-50 model, and use these features to find the  $k = 21$  nearest neighbors for each of the images in our test sets (additional details are in the supplemental material). Subsequently, we manually inspected all images and their nearest neighbors to identify duplicates. While it is difficult to know the true recall of our duplicate detection system, this procedure uncovered 150 val-IN-50k-1k (0.30%), 10 val-CUB-6k-200 (0.17%), 151 val-Places-37k-365 (0.41%), and 6 val-COCO-5k-80 (0.12%) duplicates; we will continue to improve this system and, as a result, the estimated number of duplicates may increase. In our results, we report the observed accuracy of our models; in the supplemental material, we report a conservative lower bound on accuracy by marking all duplicates as incorrect. Given the small percentage of duplicates, they do not impact our findings.

**Discussion.** Our datasets have two nice properties: public visibility and simplicity. By using publicly accessible images, the data used in our experiments is visible to everyone. To see what it looks like, the images are browsable by hashtag at <https://www.instagram.com/explore/tags/> followed by a specific hashtag; for example <https://www.instagram.com/explore/tags/brownbear> shows images tagged with #brownbear. Our data is also taken from the “wild”, essentially as-is, with minimal effort to sanitize it. This makes the dataset construction process particularly simple and transparent.

We contrast these properties with the JFT-300M dataset [17], which is not publicly visible and is the result of a proprietary collection process (“The [JFT-300M] images are labeled using an algorithm that uses a complex mixture of raw web signals, connections between web-pages and user feedback.”). Additional details describing the collection of JFT-300M have not been publicly disclosed.

Despite our efforts to make the dataset content and collection process transparent, we acknowledge that, similar to JFT-300M, it is not possible for other research groups to know exactly which images we used nor to download them en masse. Hence it is not possible for others to replicate our results at this time. However, we believe that it is better if we undertake this study and share the results with the community than to not publish the results.

## 2.2 ImageNet Datasets

In addition to the standard IN-1k dataset, we experiment with larger subsets of the full ImageNet 2011 release that contains 14.2M images and 22k labels. We construct training and validation sets that include 5k and 9k labels. For the 5k set, we use the now standard IN-5k proposed in [15] (6.6M training images). For the 9k label set, we follow the same protocol used to construct IN-5k, which

involves taking the next most frequent 4k labels and all of the associated images (10.5M training images). In all cases, we use 50 images per class for validation.

## 2.3 Models

We use residual networks with grouped convolutional layers, called ResNeXt [15]. Our experiments use ResNeXt-101  $32 \times C$ d, which has 101 layers, 32 groups, and group widths  $C$  of: 4 (8B multiply-add FLOPs, 43M parameters), 8 (16B, 88M), 16 (36B, 193M), 32 (87B, 466M), and 48 (153B, 829M). Our implementation matches [19]. We believe our results will generalize to other architectures [24–26].

**Loss Function.** In contrast to ImageNet, our Instagram datasets may contain multiple labels per image (because a user specified multiple hashtags). The average number of hashtags per image varies depending on the dataset; for instance, train-IG-1B-17k contains  $\sim 2$  hashtags per image. Our model computes probabilities over all hashtags in the vocabulary using a softmax activation and is trained to minimize the cross-entropy between the predicted softmax distribution and the target distribution of each image. The target is a vector with  $k$  non-zero entries each set to  $1/k$  corresponding to the  $k \geq 1$  hashtags for the image.

We have also experimented with per-hashtag sigmoid outputs and binary logistic loss, but obtained significantly worse results. While counter-intuitive given the multi-label data, these findings match similar observations in [16]. The successful application of sigmoid activations and logistic loss may require sophisticated label completion techniques [17] and more hyper-parameter search.

## 2.4 Pretraining Details

Our models are trained by synchronous stochastic gradient descent (SGD) on 336 GPUs across 42 machines with minibatches of 8,064 images. Each GPU processes 24 images at a time and batch normalization (BN) [27] statistics are computed on these 24 image sets. The length of the training schedule, measured in units of number-of-images-processed (i.e., minibatch size  $\times$  total SGD updates), is determined by a heuristic: we choose two training extremes (for instance, 120 epochs on 1.2e6 images and 2 epochs on 3.5e9 images) and linearly interpolate the schedule between them to set the number-of-images-processed for each experiment. Schedules for each experiment are in the supplemental material. Our ResNeXt-101  $32 \times 16$ d networks took  $\sim 22$  days to train on 3.5B images.

To set the learning rate, we follow the linear scaling rule with gradual warm-up described in [19]. We use a warm-up from 0.1 up to  $0.1/256 \times 8064$ , where 0.1 and 256 are canonical learning rate and minibatch sizes [28]. After the warm-up, the learning rate is multiplied by 0.5 at equally spaced steps, such that the total number of learning rate reductions is 20 over the course of training. The same settings are used when training on ImageNet and Instagram data, except that when training on ImageNet we use 128 GPUs in 16 machines (for a minibatch

size of 3,072) due to the smaller dataset size and we use the standard learning rate schedule that involves three equally spaced reductions by a factor of 0.1. All other initialization and training details match [19] and are summarized in the supplemental material.

### 3 Experiments

In our experiments, we pretrain convolutional networks for hashtag prediction and transfer those networks to a variety of tasks. There are two established protocols for judging the quality of a pretrained model (see [29] Sect. 3 for a discussion). Both analyze how pretraining on a *source task*, e.g. IN-1k classification, leads to gains (or losses) on a *target task*, e.g. bird recognition or object detection.

*Full network finetuning* views pretraining as sophisticated weight initialization: the success of pretraining is judged by its impact on the target task after further training the network weights in a task-specific manner (i.e. finetuning). By contrast, *feature transfer* uses the pretrained network as a feature extractor: it judges the quality of the network by how effective its features are on other tasks, without updating any of the network parameters. These protocols are two extremes of a spectrum along which the proportion of pretrained weights that are finetuned varies from all to none. We employ both protocols in our experiments; at times one is more appropriate than the other.

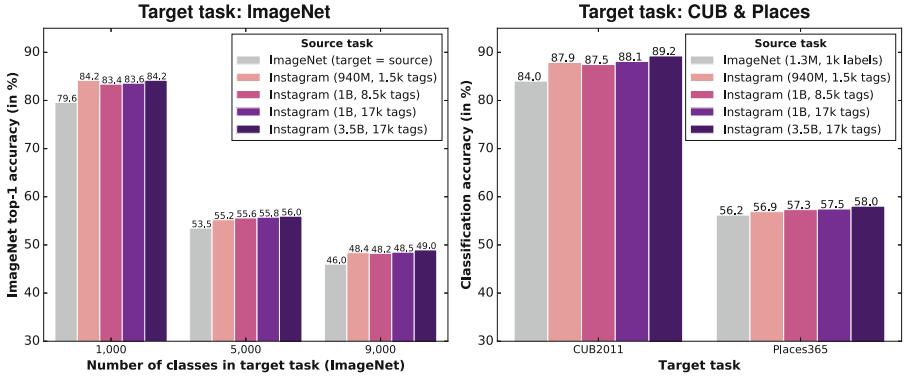
**Full network finetuning** is performed by removing the hashtag-specific fully connected classification layer from the network and replacing it with a randomly initialized classification layer with one output per class in the target task. This modified network is then trained using SGD with momentum. We select the finetuning learning rate and schedule by grid search on a proper validation set for each target task. To do this, we randomly hold out a small portion of the *training set* (see supplemental material). This practice ensures that our results on the standard validation sets are clean.

**Feature transfer** is performed by training an L2-regularized linear logistic regressor on the training data for the target task using SGD. The features produced by the pretrained network are used as input into the classifier. We train the classifier until convergence to the global optimum.

#### 3.1 Image Classification Experiments

We evaluate Instagram pretraining by measuring classification accuracies on three classification target tasks: ImageNet [30], CUB2011 [21], and Places365 [14]. We perform inference on  $224 \times 224$  center-cropped images, and study the effects of (1) the hashtag vocabulary size, (2) the training set size, (3) the amount of noise in the hashtag targets, and (4) the hashtag sampling strategy.

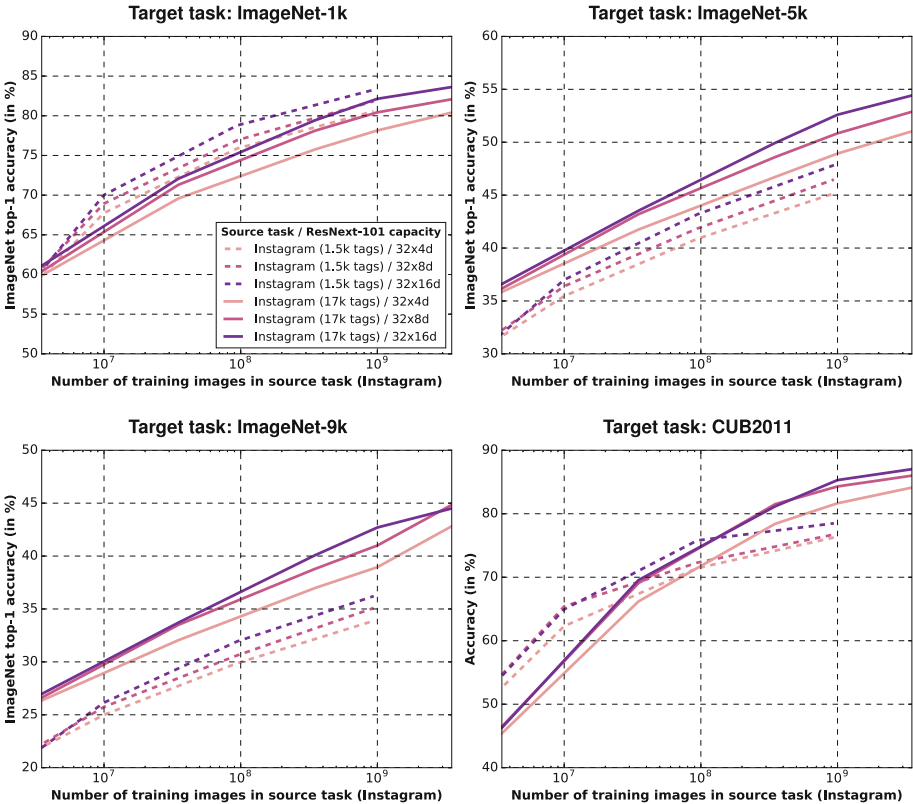
**How Does the Instagram Hashtag Set Impact Accuracy?** Our first experiment varies the Instagram hashtag sets used in pretraining (1.5k, 8.5k, vs. 17k) whilst keeping other factors constant. We compute transfer learning results as top-1 classification accuracy on five target datasets: val-IN-1k, val-IN-5k, val-IN-9k, val-CUB-200, val-Places-365. For baseline models, we use ImageNet classification as a source task: we train networks on train-IN-1k, train-IN-5k, and train-IN-9k, and evaluate them on the corresponding validation sets (finetuning is not needed in these cases). For val-CUB-200 and val-Places-365, we use train-IN-1k as the baseline source task and finetune on train-CUB-200 and train-Places-365. Full network finetuning of ResNeXt-101  $32 \times 16d$  is used for all source-target pairs in which source and target are not the same.



**Fig. 1.** small Classification accuracy of ResNeXt-101  $32 \times 16d$  pretrained on IG-1B with different hashtag vocabularies (purple bars) on IN- $\{1k, 5k, 9k\}$  (left) and CUB2011, Places365 (right). Baseline models (gray bars) are trained on IN- $\{1k, 5k, 9k\}$  (left) and IN-1k (right), respectively. Full network finetuning is used. Higher is better. (Color figure online)

Figure 1 shows that pretraining for hashtag prediction substantially improves target task accuracy: on the standard IN-1k benchmark set, a network pretrained on nearly 1B Instagram images with 1.5k hashtags achieves a state-of-the-art accuracy of 84.2%—an improvement of 4.6% over the same model architecture trained on IN-1k alone and a 1.5% boost over the prior state-of-the-art [31], which uses an optimized network architecture. The performance improvements due to Instagram pretraining vary between ImageNet tasks: on the 1k class task, the model pretrained with the IN-1k-aligned 1.5k hashtag set outperforms source networks trained on larger hashtag sets. This trend reverses as the number of target ImageNet classes increases: on 9k ImageNet target classes, the model pretrained with 17k hashtags strongly outperforms the 1.5k hashtag model. On the CUB2011 and Places365 target tasks, source models trained with the largest hashtag sets perform the best, likely, because the 17k hashtags span more objects, scenes, and fine-grained categories. These patterns are intuitive and suggest that alignment between the source and target label sets is an important factor.

We also show results in Fig. 1 using a larger 3.5B image set with 17k hashtags (dark purple bars), which performs best across all target tasks. Furthermore, following [32], we measure the *rectified* classification accuracy of this model on val-IN-1k. We present all incorrect classifications to five human annotators, asking whether or not the prediction is correct: if at least four annotators answer this question affirmatively the model’s prediction is considered correct. Whereas the IN-1M-1k model obtained a rectified top-1 accuracy of 87.5% on val-IN-1k, our IG-3.5B-17k pretrained model achieved 90.4%.



**Fig. 2.** Classification accuracies on IN-{1k, 5k, 9k} and CUB2011 target tasks as a function of the number of Instagram images used for pretraining for three network architectures (colors) and two hashtag vocabularies (dashed/solid lines). Only the linear classifier is trained on the target task. Higher is better. (Color figure online)

**How Does the Pretraining Image Set Size Impact Accuracy?** This experiment studies the relationship between the number of images used in Instagram pretraining and classification accuracy on the target task. For these experiments, when transferring to the target task we keep the pretrained network



weights fixed and only train a linear classifier for the target task. We make this choice because when the number of pretraining images is small relative to the number of target task images (e.g., 1M vs. 7M), the effect of pretraining is masked by the large amount of finetuning data (this was not the case in the previous experiment where the source task had orders of magnitude more images).

Figure 2 shows the classification accuracy on ImageNet validation sets ( $y$ -axis) as a function of the number of Instagram training images ( $x$ -axis; note the log scale) ranging from 3.5M to 3.5B images. The figure shows results for models pretrained to predict 1.5k hashtags (dashed lines) or 17k hashtags (solid lines) for ResNeXt-101 models with three different capacities (represented by different colors).<sup>1</sup> The four panels correspond to ImageNet target tasks with three different number of classes (1k, 5k, 9k) and CUB2011.

In line with prior results [16, 17], we observe near log-linear behavior: each time we multiply the amount of training data by a factor of  $x$ , we observe a fixed increase  $y$  in classification accuracy. While the scaling behavior is consistent across hashtag vocabulary sizes and models, the accuracy increase  $y$  is larger for higher-capacity networks: across all figures, the lines corresponding to ResNeXt-101  $32 \times 16d$  networks (purple) are steeper than those corresponding to  $32 \times 8d$  and  $32 \times 4d$  models. This result suggests that when training convolutional networks on billions of training images, current network architectures are prone to underfitting. We also observe log-linear scaling break down in two regimes: (1) because accuracy is bounded, endless log-linear scaling is not possible. On datasets like IN-1k and CUB2011 the ceiling effect necessarily creates sub-log-linear scaling. (2) We observe a deviation from log-linear scaling in the 1B to 3.5B image regime even without apparent ceiling effects on IN- $\{5k, 9k\}$ .

These plots also illustrate an interesting effect of the hashtag vocabulary on the transfer task accuracy. On IN-1k, networks pretrained on the target-task-aligned 1.5k hashtags outperform those trained using a larger hashtag vocabulary, because the 1.5k hashtags were selected to match the ImageNet synsets. However, as the matching between hashtag vocabulary and target classes disappears and the visual variety in the transfer task increases, networks pretrained to recognize a larger number of hashtags increasingly outperform networks pretrained on fewer hashtags: on the IN-9k transfer task, the difference in accuracy between networks trained on 1.5k and those trained on 17k hashtags is  $\sim 7\%$ . In the supplemental material, we analyze the effect of weakly supervised pretraining on recognizing individual IN-1k classes in more detail.

The highest accuracies on val-IN-1k are 83.3% (source: IG-940M-1k) and 83.6% (source: IG-3.5B-17k), both with ResNeXt-101  $32 \times 16d$ . *These results are obtained by training a linear classifier on fixed features and yet are nearly as good as full network finetuning, demonstrating the effectiveness of the feature representation learned from hashtag prediction.* These results also have low variance: we pretrained the ResNeXt-101  $32 \times 16d$  architecture of two different random samples of 1B images and then trained linear classifiers on IN- $\{1k, 5k, 9k\}$  finding a difference in top-1 accuracy of less than 0.1% in all cases.

<sup>1</sup> The maximum number of images available for the 1.5k hashtag set is 940M.

To test whether the above observations generalize to fine-grained classification, we repeated the experiments on the CUB2011 dataset, and show the results in Fig. 2, bottom right. The curves reveal that when training data is limited, the 1.5k hashtag dataset is better, but once the number of training images surpasses  $\sim 100\text{M}$ , the larger 17k hashtag dataset prevails, presumably because it represents a more diverse visual distribution with more fine-grained concepts.

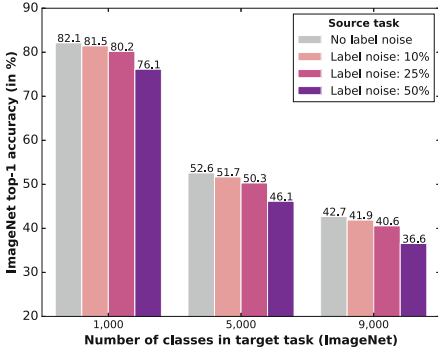
**What is the Effect of Hashtag Label Noise on Model Accuracy?** A major difference between hashtag supervision and the labels provided in datasets such as ImageNet is that hashtag supervision is inherently *noisy*: users may apply hashtags that are irrelevant to the visual content of the image, or they may have left out hashtags that would have been visually relevant [33–35]. Because an exact characterization of this label noise is difficult, instead, we investigate the effect of injecting additional label noise on the accuracy of our networks. To do so, we pretrain ResNeXt-101  $32 \times 16\text{d}$  networks on a version of IG-1B-17k in which we randomly replaced  $p\%$  of the hashtags by hashtags sampled from the marginal distribution over hashtags (excluding the tag to be replaced).

Figure 3 shows the ImageNet classification accuracy of the resulting networks for different numbers of classes at three levels,  $p$ , of artificial label noise as well as for a baseline in which no artificial label noise was added during pretraining. We only train the final linear classifier on the target task, because full finetuning may mask the damage caused by pretraining noise. *The results suggest that the networks are remarkably resilient against label noise*: a noise level of  $p = 10\%$  leads to a loss of less than 1% in classification accuracy, and at  $p = 25\%$  label noise, the reduction in accuracy is around 2%. These results suggest that label noise may be a limited issue if networks are trained on billions of images.

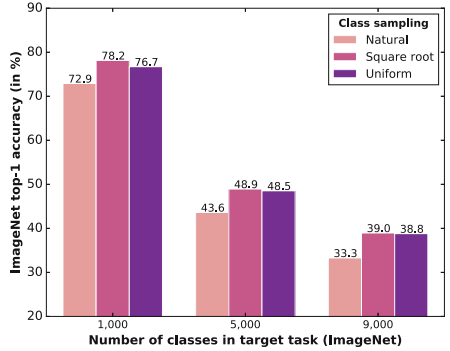
### How Does the Sampling of Pretraining Data Impact Accuracy?

Another difference between hashtag and ImageNet supervision is that, like in language modeling, hashtags are governed by a Zipfian distribution. Prior studies in language modeling found that resampling Zipfian distributions reduces the impact of the head of the word distribution on the overall training loss [36]. Motivated by this work, we perform experiments in which we evaluate three different types of data sampling in the Instagram pretraining: (1) a *natural* sampling in which we sample images and hashtags according to the distribution by which they appear on Instagram; (2) *square-root* sampling [36] in which we take the square-root of the head of the hashtag distribution, renormalize, and sample according to the resulting distribution (due to practical considerations, our implementation is slightly different; see supplemental material); and (3) *uniform* sampling in which we sample a hashtag uniformly at random, and then sample an image that has this hashtag associated to it uniformly at random [16]. (Aside from this experiment, we always pretrain on Instagram data using square-root sampling.) As before, we only train the final linear classifier on the target task.

Figure 4 displays classification accuracy as a function of the number of ImageNet classes for networks that were pretrained on IG-1B-17k using the three



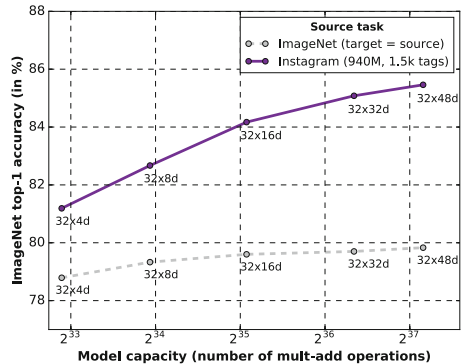
**Fig. 3.** Classification accuracy of ResNeXt-101  $32 \times 16d$ , pretrained on IG-1B-17k, on val-IN- $\{1k, 5k, 9k\}$  at three levels of injected label noise. The no-label-noise baseline is trained on the original hashtags. Only the linear classifier is trained on the target task.



**Fig. 4.** Classification accuracy of ResNeXt-101  $32 \times 4d$ , pretrained on IG-1B-17k, on val-IN- $\{1k, 5k, 9k\}$  for three different hashtag sampling strategies: natural sampling, uniform sampling, and square-root sampling. Only the linear classifier is trained on the target task.

sampling strategies. The results show that resampling of the hashtag distribution is important in order to obtain good transfer to ImageNet image-classification tasks: using uniform or square-root sampling leads to an accuracy improvement of 5 to 6% irrespective of the number of ImageNet classes in the transfer task. In line with prior results, the figure also shows that larger hashtag vocabularies lead to increasing accuracy improvements as the number of target classes grows.

**With Billions of Images, is Transfer Learning Model-Capacity Bound?** Now, we look at what happens when we train convolutional networks that are substantially larger than those typically used in recent studies (and our experiments so far). In particular, we use IG-940M-1.5k to pretrain ResNeXt-101  $32 \times 32d$  and ResNeXt-101  $32 \times 48d$ , which have  $2.4\times$  and  $4.3\times$  more add-mult FLOPs than ResNeXt-101  $32 \times 16d$ , respectively. Using these “super-sized” models improves val-IN-1k results over the  $32 \times 16d$  model from 84.2% top-1 accuracy to 85.1% and 85.4%, respectively (top-5 accuracy: from 97.2% to 97.5% and 97.6%).



**Fig. 5.** Classification accuracy on val-IN-1k using ResNeXt-101  $32 \times \{4, 8, 16, 32, 48\}d$  with and without pretraining on the IG-940M-1.5k dataset.

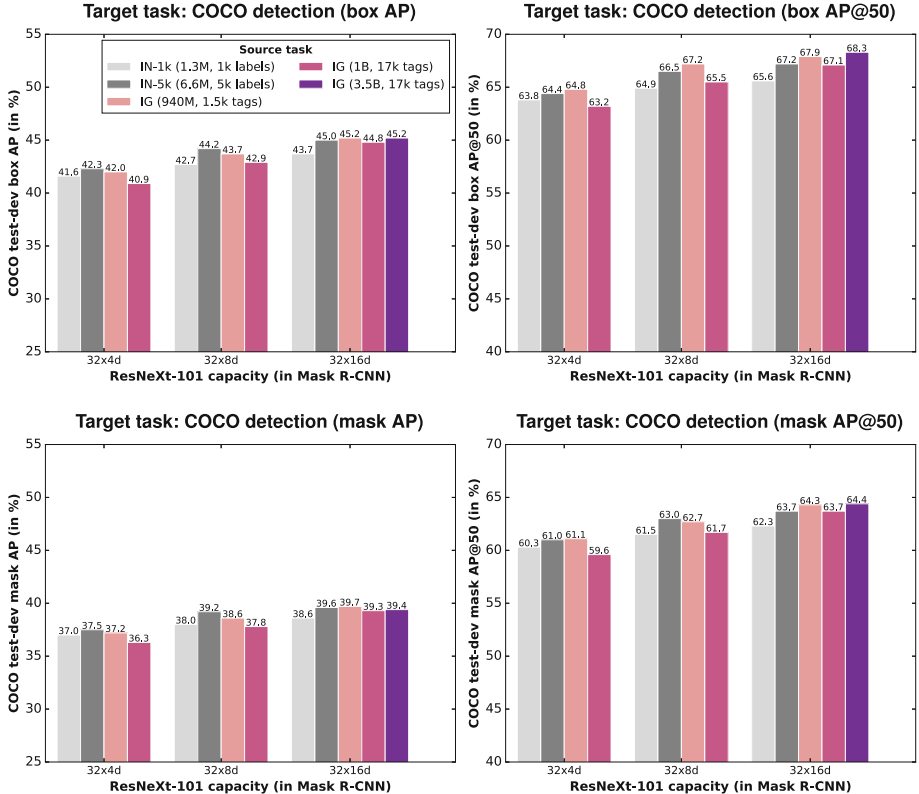
By comparison, when training from scratch on IN-1k, top-1 accuracy saturates at around 79.6% with the  $32 \times 16d$  model and does not meaningfully increase by using larger models. These results, plotted in Fig. 5, indicate that with large-scale Instagram hashtag training, transfer-learning performance appears bottlenecked by model capacity.

### 3.2 Object Detection

We have looked at target tasks that require image classification, but we are also interested in observing if pretraining on Instagram hashtag data can improve object detection and instance segmentation tasks by finetuning networks on the COCO dataset [18]. We use Mask R-CNN [4, 37] and experiment with ResNeXt-101 FPN [38] backbones of three different capacities (see Fig. 6).

We compare performance on the 2017 test-dev set using several different pretrained networks. As baselines, we use IN- $\{1k, 5k\}$  pretraining (IN-9k performs no better than IN-5k) and compare them to IG-940M-1k and IG-1B-17k. For the largest model ( $32 \times 16d$ ) we also include results for IG-3.5B-17k. We use standard settings [37] for end-to-end Mask R-CNN training with one exception: for the Instagram pretrained models we found it necessary to perform grid search for the finetuning learning rate on the validation set. We found that models pretrained on the Instagram data require finetuning learning rates that are  $\sim 4\text{--}10\times$  lower than ImageNet pretrained models (see supplemental material). This finding illustrates that finetuning recipes developed for ImageNet pretrained models do not transfer to new pretraining sets: a larger amount of pretraining data implies the need for lower finetuning learning rates.

Figure 6 shows two interesting trends. First, we observe that *when using large amounts of pretraining data, detection is model capacity bound*: with the lowest capacity model ( $32 \times 4d$ ), the gains from larger datasets are small or even negative, but as model capacity increases the larger pretraining datasets yield consistent improvements. We need even larger models to take advantage of the large-scale pretraining data. The second trend we observe comes from comparing COCO’s default AP metric (average precision averaged over intersection-over-union (IoU) overlap thresholds 0.5:0.95) to AP@50 (average precision computed at IoU threshold 0.5 only). The former emphasizes precise localization while the later allows for looser localization. We observe that the improvement over IN- $\{1k, 5k\}$  pretraining from IG-1B-1k is much larger in terms of AP@50. Thus, the gains from Instagram pretraining may be primarily due to improved object classification performance, rather than spatial localization performance. Further evidence comes from experiments with keypoint detection using Mask R-CNN, where we found that compared to IN-1k pretraining, IG-1B-1k pretraining leads to worse results (65.3% vs. 67.0% keypoint AP). These two findings suggest that pretraining for Instagram hashtag classification may reduce spatial localization performance while improving classification.



**Fig. 6.** Transfer to object detection and instance segmentation with Mask R-CNN. We compare ResNeXt-101 FPN backbones of three different capacities using a variety of source pretraining tasks. Higher is better.

## 4 Related Work

Our study is part of a larger body of work on training convolutional networks on large, weakly supervised image datasets. Sun et al. [17] train convolutional networks on the JFT-300M dataset of 300 million weakly supervised images. Our Instagram datasets are an order of magnitude larger than JFT-300M, and collecting them required much less manual annotation work (see Sect. 2.1). Due to the larger training set size and the use of better network architectures, we obtain substantially higher accuracies on transfer tasks: e.g., we obtain 85.4% top-1 accuracy on ImageNet-1k, compared to 79.2% reported in [17].

Other prior studies [16, 39] trained convolutional networks to predict words or n-grams in comments on a collection of 100 million Flickr photos and corresponding comments [40]. Word or n-gram supervision is weaker than hashtag supervision because it is less structured, as reflected in the poor transfer of features to ImageNet reported in [16]. Other work [33, 35] also trained networks to

predict hashtags on the Flickr dataset but, unlike our study, does not investigate transfer of the resulting networks to other tasks. In addition to Flickr hashtags, [41] trained hard mixture of expert models on food-related Instagram hashtags; our focus is on standard recognition networks and general hashtags. Other studies on hashtag prediction [42] do not train convolutional networks from scratch, but train linear classifiers to predict relevant hashtags from pre-defined image features. Several other works have trained models on web-scale image data for other purposes, such as face recognition [43, 44] and similarity search [45, 46], but to the best of our knowledge, we are the first to report the results of experiments that involve training convolutional networks from scratch on billions of images.

## 5 Discussion

We have attempted to explore the limits of supervised pretraining. In addition to producing state-of-the-art results on the ImageNet-1k benchmark task (85.4% single-crop, top-1 accuracy; 97.6% single-crop, top-5 accuracy) and several other vision tasks, our study has led to four important observations:

1. Our results suggests that, *whilst increasing the size of the pretraining dataset may be worthwhile, it may be at least as important to select a label space for the source task to match that of the target task*. We found that networks trained on a hashtag vocabulary that was designed to match the classes in the ImageNet-1k dataset outperformed those trained on twice as many images without such careful selection of hashtags (Fig. 2, top left). This observation paves the way for the design of “label-space engineering” approaches that aim to optimally select (weakly supervised) label sets for a particular target task. Such label-space engineering may be much more fruitful than further increasing the scale of the data on which models are trained.
2. In line with prior work [16, 17], *we observe that current network architectures are underfitting when trained on billions of images*. Whilst such underfitting does lead to very high robustness to noise in our hashtag targets (Fig. 3), our results do suggest that accuracy improvements on target tasks may be obtained by further increases of the capacity of our networks (Fig. 2). Capacity may be increased, for instance, by increasing the number of layers and the number of filters per layer of existing architectures or by mixtures-of-experts [41] (using model parallelization across GPUs). However, it is not unthinkable that some of the design choices that were made in current network architectures are too tailored to ImageNet-1k classification, and need to be revisited when training on billions of images with hashtag supervision.
3. Our results also underline the importance of increasing the visual variety that we consider in our benchmark tasks. They show that *the differences in the quality of visual features become much more pronounced if these features are evaluated on tasks with a larger visual variety*. For instance, we find that the accuracy difference between models pretrained using two different vocabularies increases as the number of target classes increases (Fig. 2): if we would have only evaluated our models on ImageNet-1k, we would have concluded

they learned visual features of similar quality, whereas results on ImageNet-9k show that one model learns substantially better features than the other. We believe evaluation on more ImageNet classes is a good step towards a more comprehensive assessment of visual recognition models.

- Results from transferring our models to object detection, instance segmentation, and keypoint detection tasks suggest that *training for large-scale hashtag prediction improves classification while at the same time possibly harming localization performance*. This opens a future direction of modifying large-scale, weakly supervised pretraining tasks to better suit the localization needs of important target tasks like detection and human pose estimation.

In closing, we reflect on the remarkable fact that training for hashtag prediction, *without the need for additional manual annotation or data cleaning*, works at all. We believe our study illustrates the potential of natural or “wild” data compared to the traditional approach of manually designing and annotating datasets.

**Acknowledgements.** We thank Matthijs Douze, Aapo Kyrola, Andrew Dye, Jerry Pan, Kevin Wilfong, Martin Englund, and Zeki Yalniz for helpful discussions and support.

## References

- Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR (2014)
- Donahue, J., et al.: DeCAF: a deep convolutional activation feature for generic visual recognition. [arXiv:1310.1531](https://arxiv.org/abs/1310.1531) (2013)
- Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional neural networks. In: ECCV (2014)
- He, K., Gkioxari, G., Dollar, P., Girshick, R.: Mask R-CNN. In: ICCV (2017)
- Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR (2015)
- Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: CVPR (2017)
- Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2D pose estimation using part affinity fields. In: CVPR (2017)
- Papandreou, G., et al.: Towards accurate multi-person pose estimation in the wild. In: CVPR (2017)
- Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: CVPR (2017)
- Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: ICCV (2015)
- Russakovsky, O., et al.: ImageNet large scale visual recognition challenge. *IJCV* **115**, 211–252 (2015)
- Agrawal, P., Girshick, R., Malik, J.: Analyzing the performance of multilayer neural networks for object recognition. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014 Part VII. LNCS, vol. 8695, pp. 329–344. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-10584-0\\_22](https://doi.org/10.1007/978-3-319-10584-0_22)
- Huh, M., Agrawal, P., Efros, A.: What makes ImageNet good for transfer learning? [arXiv:1608.08614](https://arxiv.org/abs/1608.08614) (2016)

14. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: a 10 million image database for scene recognition. *PAMI* **40**, 1452–1464 (2017)
15. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: *CVPR* (2017)
16. Joulin, A., van der Maaten, L., Jabri, A., Vasilache, N.: Learning visual features from large weakly supervised data. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *ECCV 2016 Part VII. LNCS*, vol. 9911, pp. 67–84. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-46478-7\\_5](https://doi.org/10.1007/978-3-319-46478-7_5)
17. Sun, C., Shrivastava, A., Singh, S., Gupta, A.: Revisiting unreasonable effectiveness of data in deep learning era. In: *Proceedings of ICCV* (2017)
18. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014 Part V. LNCS*, vol. 8693, pp. 740–755. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-10602-1\\_48](https://doi.org/10.1007/978-3-319-10602-1_48)
19. Goyal, P., et al.: Accurate, large minibatch SGD: training ImageNet in 1 hour. In: [arXiv:1706.02677](https://arxiv.org/abs/1706.02677) (2017)
20. WordNet: About WordNet (2010). <http://wordnet.princeton.edu>
21. Welinder, P., et al.: Caltech-UCSD Birds 200. Technical report, Caltech (2010)
22. Gordo, A., Almazan, J., Revaud, J., Larlus, D.: Deep image retrieval: learning global representations for image search. In: [arXiv:1604.01325](https://arxiv.org/abs/1604.01325) (2016)
23. Tolias, G., Sicre, R., Jegou, H.: Particular object retrieval with integral max-pooling of CNN activations. In: *ICLR* (2016)
24. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *CVPR* (2016)
25. Huang, G., Liu, Z., Weinberger, K., van der Maaten, L.: Densely connected convolutional networks. In: *CVPR*. (2017)
26. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.: Inception-V4, inception-ResNet and the impact of residual connections on learning. In: [arXiv:1602.07261](https://arxiv.org/abs/1602.07261) (2016)
27. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. In: *ICML* (2015)
28. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: surpassing human-level performance on ImageNet classification. In: *ICCV* (2015)
29. Pathak, D., Girshick, R., Dollár, P., Darrell, T., Hariharan, B.: Learning features by watching objects move. In: *CVPR* (2017)
30. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: *CVPR* (2009)
31. Zoph, B., Vasudevan, V., Shlens, J., Le, Q.: Learning transferable architectures for scalable image recognition. In: [arXiv:1707.07012](https://arxiv.org/abs/1707.07012) (2017)
32. Stock, P., Cisse, M.: Convnets and ImageNet beyond accuracy: explanations, bias detection, adversarial examples and model criticism. In: [arXiv:1711.11443](https://arxiv.org/abs/1711.11443) (2017)
33. Izadinia, H., Russell, B., Farhadi, A., Hoffman, M., Hertzmann, A.: Deep classifiers from image tags in the wild. In: *Multimedia COMMONS* (2015)
34. Misra, I., Zitnick, C.L., Mitchell, M., Girshick, R.: Seeing through the human reporting bias: visual classifiers from noisy human-centric labels. In: *CVPR* (2016)
35. Veit, A., Nickel, M., Belongie, S., van der Maaten, L.: Separating self-expression and visual content in hashtag supervision. In: [arXiv 1711.09825](https://arxiv.org/abs/1711.09825) (2017)
36. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: *NIPS* (2013)
37. Girshick, R., Radosavovic, I., Gkioxari, G., Dollár, P., He, K.: Detectron (2018). <https://github.com/facebookresearch/detectron>



38. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: CVPR (2017)
39. Li, A., Jabri, A., Joulin, A., van der Maaten, L.: Learning visual n-grams from web data. In: Proceedings of ICCV (2017)
40. Thomee, B., et al.: YFCC100M: the new data in multimedia research. *Commun. ACM* **59**(2), 64–73 (2016)
41. Gross, S., Ranzato, M., Szlam, A.: Hard mixtures of experts for large scale weakly supervised vision. In: CVPR (2017)
42. Denton, E., Weston, J., Paluri, M., Bourdev, L., Fergus, R.: User conditional hashtag prediction for images. In: Proceedings of KDD, pp. 1731–1740 (2015)
43. Schroff, F., Kalenichenko, D., Philbin, J.: FaceNet: a unified embedding for face recognition and clustering. In: CVPR (2015)
44. Taigman, Y., Yang, M., Ranzato, M., Wolf, L.: Web-scale training for face identification. In: CVPR (2015)
45. Johnson, J., Douze, M., Jégou, H.: Billion-scale similarity search with GPUs. arXiv preprint [arXiv:1702.08734](https://arxiv.org/abs/1702.08734) (2017)
46. Stewénius, H., Gunderson, S.H., Pilet, J.: Size matters: exhaustive geometric verification for image retrieval accepted for ECCV 2012. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012 Part II. LNCS, pp. 674–687. Springer, Heidelberg (2012). [https://doi.org/10.1007/978-3-642-33709-3\\_48](https://doi.org/10.1007/978-3-642-33709-3_48)