Learning with Neighbor Consistency for Noisy Labels

Supplementary Material

Table 4. List of network hyperparameters used in our experiments.

	CIFAR-{10, 100}	mini-{Red, Blue}	mini-Webvision	Clothing1M
Opt.	SGD			
Momentum	0.9			
Batch	256	128	256	128
LR	0.1	0.1	0.1	0.002
LR Sch.	cosine decay with linear warmup			
Warmup	5			
Epochs	250	130	130	80
Weight Dec.	$5e-4$	$5e-4$	$1e-3$	$1e-3$
Arch.	ResNet-18		ResNet-50	

A. Training details

Implementation details. We use the ResNet-18 and -50 architectures [14] in our experiments. We follow the same protocol as ELR [27], in CIFAR experiments. For Clothing1M, we follow the work of [23] and fine-tune a pre-trained ResNet-50 for 80 epochs, where each epoch contains 1000 mini-batches. Table 4 lists the network hyperparameters used to train the network throughout our experiments. We train the network with the typical dot-product linear classifier $h_W()$ in all datasets except for mini-WebVision and WebVision. For the mini-WebVision and WebVision experiments, we follow the work of [31] and use a *cosine classifier* for $h_W()$. Cosine classifier is also a linear classifier, however, the features and the classifier weights are ℓ_2-normalized unlike the dot-product classifier.

We employ random crop augmentation during training in all experiments and resize images to 224 × 224 pixels. For experiments with CIFAR, we use 32×32 images and reduce the strides. We trained each model on a single Nvidia V100 GPU, and will release all code upon acceptance.

NCR hyperparameters. We sweep over the NCR hyperparameters α, k and e, and choose a set of hyperparameter based on the validation set accuracy on CIFAR-{10, 100} and Clothing1M. This hyperparameter sweep is done for each noise ratio separately. Since mini-ImageNet-{Red, Blue} does not contain a held-out validation set , we create a *held-out* set from the mini-ImageNet-Red dataset which comprises the (clean) examples from the 0% noise dataset that do not appear in the datasets with 20%, 40% or 80% noise. The *held-out* set allows us to choose hyperparameters without overfitting on the final evaluation set. We use the same hyperparameters on mini-ImageNet-Blue as well. Table 5 shows the list of hyperparameters for each dataset.

Table 5. List of NCR hyperparameters used in our experiments.

	mini-ImageNet				CIFAR-10	CIFAR-100	WebVision	Clothing1M
	0%	20%	40%	80%				
α	0.7	0.7	0.7	0.5	0.1	0.1	0.5	0.9
k	50	1	1	1	10	10	10	1
e	100	50	50	0	50	200	0	40

Table 6. Effect of the batch size on our proposed NCR method, on the WebVision dataset containing 1000 classes.

Batch Size	256	512	1024	2048
Accuracy	73.9	75.0	75.7	75.6

B. Dataset details

Mini-ImageNet-Red contains 50 000 training examples and 5 000 validation examples. The noisy images are retrieved by text-to-image and image-to-image search. They come from an open vocabulary outside of the set of classes in the training set. Depending on the noise ratio, a subset of clean images are replaced by the noisy images to construct the training set. **Mini-ImageNet-Blue** contains 60 000 training examples. The validation set is the same as mini-ImageNet-Red. The noise in mini-ImageNet-Blue is synthetic. The label of each example is independently and uniformly changed according to a probability. The noisy examples come from a fixed vocabulary, *i.e.* their true label belongs to another class in the training set. **WebVision** contains 2.4M images and 1000 classes. Images are collected from the web, using the Google and Flickr search engines. The data is *imbalanced*, meaning each class contains a different number of training examples. **Mini-Webvision** contains a subset of the original Webvision dataset [26]. It contains only the first 50 classes of the Google image subset. This corresponds to 65 944 training images. The validation set contains 2 500 images corresponding to the 50 training classes. **Clothing1M** [44] is a large-scale dataset containing 1 million images and 14 categories. Images are collected from the web, and the noisy labels are assigned based on the surrounding text. We do not use the clean training subset with human-verified labels. We follow the existing protocol [24] and fine-tune a ResNet-50 model which is pre-trained on ImageNet.

C. Effect of batch size and number of labels

We use the WebVision dataset to to analyse the effect of the batch size for large label vocabularies. WebVision is a large-scale dataset containing 2.4M images and 1K classes. We report the accuracy for different batch sizes in Table 6.

The accuracy increases with batch size, plateauing after the batch size is more than the number of classes. This shows that NCR requires the batch size to be approximately the number of classes, but that much bigger batch sizes are not required. We use a batch size of 1024 for WebVision. The other training hyperparameters remain the same as mini-Webvision on Table 4.