Robust and On-the-fly Dataset Denoising for Image Classification

Jiaming Song[1], Yann Dauphin[2], Michael Auli[3], and Tengyu Ma[1]

[1] Stanford University
{tsong, tengyuma}@stanford.edu
[2] Google Brain
yann@dauphin.io
[3] Facebook AI Research
michaelauli@fb.com

Abstract. Memorization in over-parameterized neural networks could severely hurt generalization in the presence of mislabeled examples. However, mislabeled examples are hard to avoid in extremely large datasets collected with weak supervision. We address this problem by reasoning counterfactually about the loss distribution of examples with uniform random labels had they were trained with the real examples, and use this information to remove noisy examples from the training set. First, we observe that examples with uniform random labels have higher losses when trained with stochastic gradient descent under large learning rates. Then, we propose to model the loss distribution of the counterfactual examples using only the network parameters, which is able to model such examples with remarkable success. Finally, we propose to remove examples whose loss exceeds a certain quantile of the modeled loss distribution. This leads to On-the-fly Data Denoising (ODD), a simple yet effective algorithm that is robust to mislabeled examples, while introducing almost zero computational overhead compared to standard training. ODD is able to achieve state-of-the-art results on a wide range of datasets including real-world ones such as WebVision and Clothing1M.

1 Introduction

Over-parametrized deep neural networks have remarkable generalization properties while achieving near-zero training error [45]. However, the ability to fit the entire training set is highly undesirable, as a small portion of mislabeled examples in the dataset could severely hurt generalization [45, 1]. Meanwhile, an exponential growth in training data size is required to linearly improve generalization in vision [37]; this progress could be hindered if there are mislabeled examples within the dataset.

Mislabeled examples are to be expected in large datasets that contain millions of examples. Web-based supervision produces noisy labels [21, 26] whereas

---

Work done at Facebook AI research.

human labeled datasets sacrifice accuracy for scalability [19]. Therefore, algorithms that are robust to various levels of mislabeled examples are warranted in order to further improve generalization for very large labeled datasets.

In this paper, we are motivated by the observation that crowd-sourcing or web-supervision could have multiple disagreeing sources; in such cases, noisy labels could exhibit higher conditional entropy than the ground truth labels. Since the information about the noisy labels (such as the amount of noise) is often scarce, we pursue *a general approach* by counterfactual reasoning of the behavior of noisy examples with high conditional entropy. Specifically, we reason about the *counterfactual* case of how examples with uniform random noise would behave *had they appeared in the training dataset*, without actually training on such labels. If a real example has higher loss than what most counterfactual examples with uniform random noise would have, then there is reason to believe that this example is likely to contain a noisy label; removing this example would then improve performance on a clean test set.

To reason about the *counterfactual loss distribution* of examples with uniform random noise, we first show that training residual networks with *large learning rates* will create a significant gap between the losses of clean examples and noisy examples. The distribution of training loss over clean examples decrease yet that of the uniformly noisy examples does not change, regardless of the proportion of noisy examples in the dataset. Based on this observation, we propose a distribution that simulates the loss distribution of uniform noisy examples based only on the network parameters. Reasonable thresholds can be derived from percentiles of this distribution, which we can then utilize to denoise the dataset. This is critical in real-world applications, because prior knowledge about the distribution of label noise is often scarce; even if we have such information (such as transition matrices of label noise), algorithms that specifically utilize this information are not scalable when there are thousands of labels.

We proceed to propose *On-the-fly Data Denoising* (ODD, see Figure 1), a simple and robust method for training with noisy examples based on the implicit regularization effect of stochastic gradient descent. First, we train residual networks with large learning rate schedules and use the resulting losses to separate clean examples from mislabeled ones. This is done by identifying examples whose losses exceed a certain threshold. Finally, we remove these examples from the dataset and continue training until convergence. ODD is a general approach that can be used to train clean dataset as well as noisy datasets with almost no modifications.

Empirically, ODD performs favorably against previous methods in datasets containing *real-world noisy examples*, such as WebVision [21] and Clothing1M [41]. ODD also achieves equal or better accuracy than the state-of-the-art *on clean datasets*, such as CIFAR and ImageNet. We further conduct ablation studies to demonstrate that ODD is robust to different hyperparameters and artificial noise levels. Qualitatively, we demonstrate the effectiveness of ODD by detecting mislabeled examples in the "clean" CIFAR-100 dataset without any supervision other than the training labels (Figure 2). These results suggest that we can use
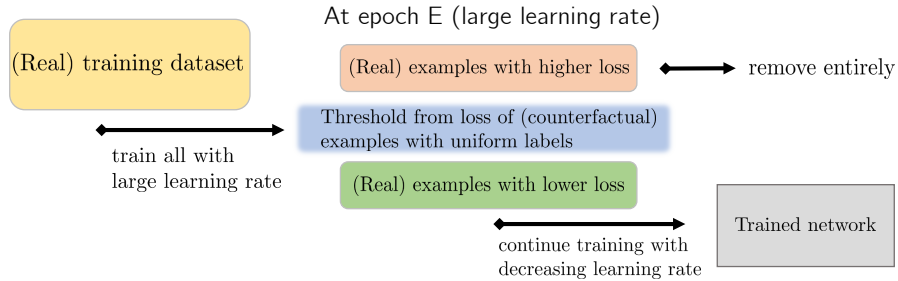
Fig. 1. Pipeline of our method. We utilize the implicit regularization effect of SGD to (counterfactually) reason the loss distribution of examples with uniform label noise. We remove examples that have loss higher than the threshold and train on the remaining examples. There is no assumption that the dataset has to contain uniformly random labels (thus such labels are "counterfactual"); we empirically validate our method on real-world noisy datasets.

Fig. 2. Mislabeled examples in the CIFAR-100 training set detected by ODD.

ODD in both clean and noisy datasets with minimum computational overhead to the training algorithm.

2 Problem setup

The goal of supervised learning is to find a function $f \in \mathcal{F}$ that describes the probability of a random label vector $Y \in \mathcal{Y}$ given a random input vector $X \in \mathcal{X}$, which has underlying joint distribution $P(X, Y)$. Given a loss function $\ell(\mathbf{y}, \hat{\mathbf{y}})$, one could minimize the average of ℓ over P:

$$\mathcal{R}(f) = \int_{\mathcal{X} \times \mathcal{Y}} \ell(\mathbf{y}, f(\mathbf{x})) \, \mathrm{d}P(\mathbf{x}, \mathbf{y}), \tag{1}$$

The joint distribution $P(X, Y)$ is usually unknown, but we could gain access to its samples via a potentially noisy labeling process, such as crowdsourcing [19] or web queries [21].

We denote the training dataset with N examples as $\mathcal{D} = (\mathbf{x}_i, \mathbf{y}_i)_{i \in [N]} = \mathcal{G} \cup \mathcal{B}$. $\mathcal{G}$ represents correctly labeled (clean) examples sampled from $P(X, Y)$. $\mathcal{B}$ represents mislabeled examples that are not sampled from $P(X, Y)$, but from another distribution $Q(X, Y)$. We assume that $\mathcal{G} \cap \mathcal{B} = \varnothing$, as a sample should not be both correctly labeled and mislabeled.

We aim to learn the function f from $\mathcal{D}$ *without knowledge about $\mathcal{B}$, $\mathcal{G}$ or their statistics* (e.g. the amount of mislabeled examples $|\mathcal{B}|$). A typical approach is to pretend that $\mathcal{B} = \varnothing$ — i.e., all examples are i.i.d. from $P(X, Y)$ — and minimize the empirical risk:

$$\hat{\mathcal{R}}(f) = \frac{1}{N} \sum_{i=1}^{N} \ell(\mathbf{y}, f(\mathbf{x})).$$

If $\mathcal{B} = \varnothing$ is indeed true, then the empirical risk converges to the population risk: $\hat{\mathcal{R}}(f) \rightarrow \mathcal{R}(f)$ as $N \rightarrow \infty$. However, if $\mathcal{B} \neq \varnothing$, then $\hat{\mathcal{R}}(f)$ is no longer an unbiased estimator of $\mathcal{R}(f)$. Moreover, when the hypothesis class $\mathcal{F}$ contains large neural nets with the number of parameters exceeding N, the empirical risk minimizer could fit the entire training dataset, including the mislabeled examples [45]. Overfitting to wrong labels empirically causes poor generalization. For example, training CIFAR-10 with 20% of uniformly mislabeled examples and a residual network gives a test error of 11.5%, which is significantly higher than the 4.25% error obtained with training on the clean examples.

2.1   Entropy-based Assumption over Noisy Labels

Therefore, if we were able to identify the clean examples belonging to $\mathcal{G}$, we could vastly improve the generalization on $P(X, Y)$; this requires us to provide valid prior assumptions that could distinguish clean examples from mislabeled ones. We note that these assumptions have to be general enough so as to *not depend on additional assumptions specific to each dataset*. For example, knowledge about noise transition matrices is not allowed.

We assume that for any example $\boldsymbol{x} \in \mathcal{X}$, the entropy of the clean label distribution is smaller than that of the noisy label distribution:

$$\mathcal{H}(P(Y|X = \boldsymbol{x})) < \mathcal{H}(Q(Y|X = \boldsymbol{x})) \quad \forall \boldsymbol{x} \in \mathcal{X} \tag{2}$$

where the randomness of labeling $Q(Y|X)$ could arise from noisy labelings, such as Mechanical Turk [19]. Let ℓ be the cross entropy loss, then the ERM objective is essentially trying to minimize the KL divergence between the empirical conditional distribution (denoted as $\hat{P}(\boldsymbol{y}|\boldsymbol{x})$) and the conditional distribution parametrized by our model (denoted as $p_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x})$):

$$\mathbb{E}_{\hat{P}(\boldsymbol{y}|\boldsymbol{x})}[-\log p_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x})] = \mathcal{H}(\hat{P}(\boldsymbol{y}|\boldsymbol{x})) + D_{\mathrm{KL}}(\hat{P}(\boldsymbol{y}|\boldsymbol{x})\|p_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x})) \tag{3}$$

which is minimized as $D_{\mathrm{KL}} \rightarrow 0$; in this case, the cross entropy loss is higher if $\hat{P}$ has higher entropy, which suggests that the mislabeled examples are likely to have higher loss than correct ones.

3   Denoising datasets on-the-fly with Counterfactual Thresholds

In the following section, we study the behavior of samples with uniformly random label noise; this allows us to reason about their loss distribution *counterfactually*,

and develop suitable thresholds to remove noisy examples that appear in the training set. We denote the conditional distribution of uniformly random label noise as $Q_U(Y|X)$, which is the distribution that maximizes entropy; therefore, any real-world noise distribution $Q(Y|X)$ will have smaller entropy than Q_U.

While it is unreasonable to assume that the label noise is uniformly random in practice, we do not make such assumption over our training set. Instead, we reason about the following counterfactual case:

> *Had the training set contained some examples with uniform random labels, can we characterize the loss distribution of these examples?*

We illustrate how such a counterfactual analysis allows simple and practical algorithms that work even under real-world noisy datasets. The following sections are organized as follows:

1. First, we show that when training ResNets via SGD with large learning rates[4], the training loss of uniform noisy labels and clean labels can be clearly separated. As the learning rate becomes smaller, however, the model would begin to overfit the noisy labels, leading to poor generalization properties.
2. Next, we propose an approach to model the (counterfactual) loss distribution *by only looking at the weights of the network*. We empirically show that this does not depend on the type or the amount of noisy labels in the dataset, making this approach generalize well to various counterfactual scenarios (such as different portions of uniform random labels in the dataset).
3. Finally, we can simply remove all examples that perform worse than a certain percentile of the counterfactual distribution. Since higher entropy examples tend to have higher loss than lower entropy ones, the samples we remove are more likely to be more noisy. In Fig. 2, we empirically demonstrate that the proposed threshold identifies mislabeled samples in CIFAR-100 even without any additional supervision, validating our assumption.

3.1   Separating mislabeled examples via SGD

First, we find that training the model with stochastic gradient descent (SGD) with large learning rates (e.g. 0.1) will result in significant discrepancy between the loss statistics of the clean examples and mislabeled examples. To illustrate this, we consider training deep residual networks on CIFAR-100 and ImageNet with different percentages of uniform label noise (20% and 40%), but with large learning rates (close to 0.1), and at specific epochs, we plot the histogram of the loss for each example in Fig. 3.

From the histograms, we may draw two key observations that are useful for cleaning the dataset without knowledge about the exact percentage of noise:

---

[4] Here, "large learning rates" refer to ones that are close to initial learning rates in practice; in standard ResNet training [10], the initial learning rate is 0.1, so we consider it "large"; we use the default momentum parameter of 0.9 for all cases discussed.
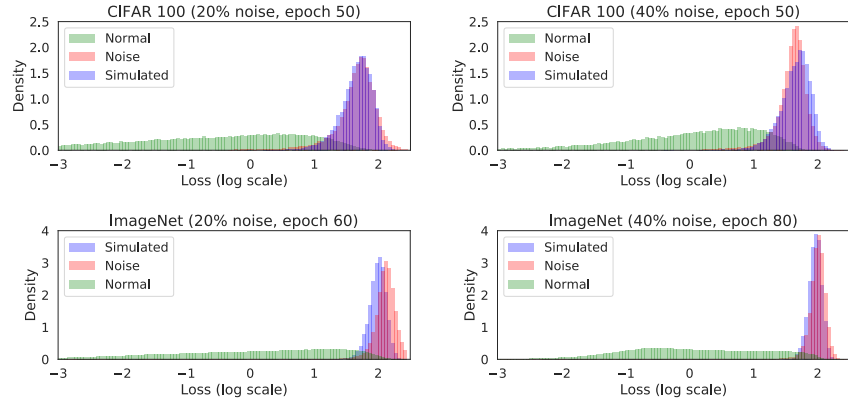
Fig. 3. Histogram of the distributions of losses, where "normal", "noise", and "simulated" denote (real) examples with clean labels, (real) examples with uniform random labels and the counterfactual model $q_n(\ell)$ respectively. $q_n(\ell)$ matches the loss distribution of noisy examples, which have higher loss than clean ones; $q_n(\ell)$ depends only on the network parameters.

– First, the loss distributions of the clean examples and the mislabeled ones have *notable statistical distance.*
– Second, it seems that the loss distribution of the uniform labeled examples are relatively stable, and *does not depend much on the amount of uniform random noise in the training set.*

These observations are consistent with those in [45], as the network starts to fit mislabeled examples when learning rate decrease further; decreasing learning rate is crucial for achieving better generalization on clean datasets.

The working of the implicit regularization of stochastic gradient descent is by and large an open question that attracts much recent attentions [28, 22, 6, 27]. Empirically, it has been observed that large learning rates are beneficial for generalization [18]. Chaudhari and Soatto [4] have argued that SGD iterates converge to limit cycles with entropic regularization proportional to the learning rate and inversely proportional to batch size. Training with large learning rates under fixed batch sizes could then encourage solutions that are more robust to large random perturbations in the parameter space and less likely to overfit to mislabeled examples, which would partially explain our observations in Fig 3.

Given these empirical and theoretical evidences on the benefits of large learning rates on generalization, we propose to classify correct and mislabeled examples through the loss statistics, and achieve better generalization by removing the examples that are potentially mislabeled.

3.2   Thresholds that classify mislabeled examples

The above observation suggests that it is possible to distinguish clean and noisy examples *via a threshold over the loss value*. In principle, we can claim an example is noisy if its loss value exceeds a certain threshold; by removing the noisy labels from the training set, we could then improve generalization performance on clean validation sets.

However, to improve generalization in practice, one critical problem is to select a reasonable threshold for classification. High thresholds could include too many examples from $\mathcal{B}$ (the mislabeled set), whereas low thresholds could prune too many examples from $\mathcal{G}$ (the clean set); reasonable thresholds should also adapt to different ratios of mislabeled examples, which could be unknown to practitioners.

If we are able to characterize the loss of $Q_U(Y|X)$ (the highest entropy distribution), we can select a reasonable threshold from this loss as any example having higher loss is likely to have high entropy labels (and is possibly mislabeled). From Fig. 3, the loss distribution for $\mathcal{B}$ is relatively stable with different ratios of $|\mathcal{B}|/|\mathcal{D}|$; examples in $\mathcal{B}$ are making little progress when learning rate is large. This suggests a threshold selecting criteria that is *independent of the amount of mislabeled examples in the dataset.*

We propose to characterize the loss distribution of (counterfactual) uniform label noise via the following procedure:

$$l = -\tilde{\mathbf{y}}_k + \log \left(\sum_{i \in [N]} \exp(\tilde{\mathbf{y}}_i) \right) \tag{4}$$

$$\tilde{\mathbf{y}} = \mathrm{fc}(\mathrm{relu}(\tilde{\mathbf{x}})), \tilde{\mathbf{x}} \sim \mathcal{N}(0, I), k \sim \mathrm{Uniform}\{0, \dots, K\}$$

We denote this counterfactual distribution model as $q_n(l)$. $q_n(l)$ tries to simulate the behavior of the model (and the loss distribution) with several components.

- k represents a random label from K classes. This simulates the case where $Q(Y|X)$ has the highest entropy, i.e. uniformly random.
- $\mathrm{fc}(\cdot)$ is the final (fully connected) layer of the network and $\mathrm{relu}(\tilde{\mathbf{x}}) = \max(\tilde{\mathbf{x}}, \mathbf{0})$ is the Rectified Linear Unit. This simulates the behavior at the last layer of the network outputs $\tilde{\boldsymbol{y}}$.
- $\tilde{\mathbf{x}} \sim \mathcal{N}(0, I)$ suggests that the inputs to the last layer has an identity covariance; the scale of the covariance could result from well-conditioned objectives defined via deep residual networks [10], batch normalization [15] and careful initialization [11].

We qualitatively demonstrate the validity of our characterization on CIFAR-100 and ImageNet datasets in Fig. 3, where we plot the histogram of the $q_n(l)$ distribution for CIFAR-100 and ImageNet (our estimates), and compare then with the empirical distribution of the loss of uniform noisy labeled examples (ground truth). The similarities between the noisy loss distribution and simulated loss distribution $q_n(l)$ demonstrate that an accurate characterization of the loss distribution can be made *without prior knowledge of the mislabeled examples.*

To effectively trade-off between precision (correctly identifying noisy examples) and recall (identifying more noisy examples), we define a threshold via the p-th percentile of $q_n(l)$ using the samples generated by Equation 4; it relates to approximately how much examples in $\mathcal{B}$ we would retain if $Q(Y|X)$ is uniform. In Section 4.5, we show that this method is able to identify different percentages of uniform label noise with high precision.

3.3   A Practical Algorithm for Robust Training

We can utilize this to remove examples that might harm generalization, leading to *On-the-fly Data Denoising* (ODD), a simple algorithm that is robust to mislabeled examples. First, we train all the samples using a relatively large learning rate (e.g. 0.1), which should favor learning clean samples than noisy ones. Next, we use the threshold to cutoff samples that have large loss values, which we deem as having been mislabeled. Finally, we train the remaining samples using the original ERM procedure.

Algorithm 1 On-the-fly Data Denoising

Input: dataset $\mathcal{D}$, model f_θ, percentile p, epoch E, learning rate schedule $\eta(t)$.
for $e = 1 \ldots E$ **do**
    Train on $\mathcal{D}$ with learning rate $\eta(e)$.
end for
$T_p = p$-th percentile of $q_n(\ell)$ in Eq. (4)
Remove "noisy" data according to threshold: $\mathcal{G} = \{(\boldsymbol{x}, \boldsymbol{y}) \mid \ell(\boldsymbol{y}, f_\theta(\boldsymbol{x})) < T_p\}$
for $e = E + 1 \ldots$ **do**
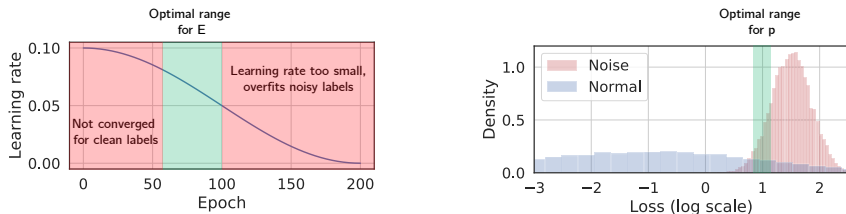   Train on $\mathcal{G}$ with learning rate $\eta(e)$.
end for

Fig. 4. Hyperparameter selection. (Left) Cosine learning rate schedule across epochs that is typical in ResNet training; we wish to select E before learning rate becomes small, and after training over clean labels have converged. (Right) Histogram of the losses; we wish to select p that does not remove too many clean data, but also removes as many (conterfactually) noisy data as possible.

Hyperparameter selection ODD introduces two hyperparameters: E determines the amount of training that separates clean examples from noisy ones; p determines T_p that specifies the trade-off between less noisy examples and more clean examples. We do not explicitly estimate the portion of noise in the dataset, nor do we assume any specific noise model. Moreover, ODD is compatible with existing practices for learning rate schedules, such as stepwise [10] or cosine [24].

In Fig. 4 we demonstrate and discuss how to choose the hyperparameters E and p. For E, we wish to perform ODD operation at a point not too early (to allow enough time for training on clean labels to converge) and not too late (to prevent overfitting noisy labels with small learning rates). For p, we wish to trade-off between keeping as much clean data as possible and removing counterfactually noisy data; selecting $p \in [1, 30]$ typically works for our case.

4 Experiments

We evaluate our method extensively on several clean and noisy datasets including CIFAR-10, CIFAR-100, ImageNet [34], WebVision [21] and Clothing1M [41]. Our experiments consider datasets that are clean, have artificial noise (in CIFAR-10, CIFAR-100 and ImageNet), or have inherent noise from web-supervision (as in the case of WebVision and Clothing1M).

Table 1. Validation accuracy (in percentage) with uniform label noise. $\star$ denotes methods trained with knowledge of 1000 additional clean labels

	CIFAR-10			CIFAR-100		
% mislabeled	0	20	40	0	20	40
ERM	96.3	88.5	84.4	81.6	69.6	55.7
mixup	97.0	93.9	91.7	81.4	71.2	59.4
GCE	-	89.9	87.1	-	66.8	62.7
Luo	96.2	**96.2**	94.9	81.4	80.6	74.2
Ren*	-	-	86.9	-	-	61.4
MentorNet*	-	92.0	89.0	-	73.0	68.0
ODD	96.2	94.7	92.8	81.8	77.2	72.4
ODD + *mixup*	**97.2**	**95.6**	**95.5**	**82.5**	**79.1**	**76.5**

4.1 CIFAR-10 and CIFAR-100

We first evaluate our method on the CIFAR-10 and CIFAR-100 datasets. We train the wide residual network architecture (WRN-28-10) in [44] for 200 epochs with a minibatch size of 128, momentum 0.9 and weight decay 5×10^{-4}. We set $E = 75$ (total number of epochs is 200) and $p = 10$ in our experiments.

Fig. 5. Examples with label "leopard" that are classified as mislabeled.

Fig. 6. Random CIFAR-100 examples that are classified as mislabeled.

Input-Agnostic Label Noise We first consider label noise that are agnostic to inputs. Following [45], We randomly replace 0%/20%/40% of the training labels to uniformly random ones, and evaluate generalization error on the clean validation set. We compare with the following baselines: Empirical Risk Minimization (ERM, Eq. 1, [8]), MENTORNET [16], REN [32], *mixup* [46], Generalized Cross Entropy (GCE, [47]) and LUO [25], a strong baseline which regularizes the Jacobian of the network. In particular, LUO would spend about two times the computational effort when performing one iteration due to back-propagation through the Jacobian. We also consider using *mixup* training after pruning noisy examples with ODD. From the results in Table 1, ODD + *mixup* outperforms all other algorithms (except for LUO).

Mislabeled examples in CIFAR-100 To demonstrate that ODD can identify mislabeled examples even in practice, we display the examples in CIFAR-100 training set for which our ODD methods identify as noise across 3 random seeds. One of the most common case is the "leopard" label; in fact, 21 "leopard" examples in the training set are perceived as hard, and we show some of them in Fig. 5. It turns out that a lot of the "leopard" examples contains images that clearly contains tigers and black panthers (CIFAR-100 has a label corresponding to "tiger", and only 2 out of 100 "leopard" test images are black panthers). We also demonstrate random examples from the CIFAR-100 that are identified as noise in Fig. 6. The examples identified as noise often contains multiple or ambiguous objects. We include more results in Section A.2.

Table 2. Results on non-homogeneous labels.

% samples removed	CIFAR-50			CIFAR-20		
	0	20	40	0	20	40
ERM	78.5	77.9	77.5	86.4	85.1	84.4
ODD	**79.0**	**78.6**	**78.1**	**86.6**	**85.4**	**84.7**

Table 3. Top-1 (top-5) accuracy on ImageNet.

% mislabeled	0	20	40
ERM	76.61 (93.23)	73.77 (91.49)	71.39 (89.48)
Luo	76.73 (93.31)	75.17 (92.26)	73.19 (91.01)
MentorNet	-	-	65.1 (85.9)
ODD ($p = 10$)	**76.71** (93.23)	**74.95** (92.11)	**72.49** (90.75)

Non-Homogeneous Labels We evaluate ERM and ODD on a setting without mislabeled examples, but the ratio of classes could vary. To prevent the model from utilizing the number of examples in a class, we combine multiple classes of CIFAR-100 into a single class, creating the CIFAR-20 and CIFAR-50 tasks. In CIFAR-50, we combine an even class with an odd class while we remove $c\%$ of the examples in the odd class. In CIFAR-20, we combine 5 classes in CIFAR-100 that belong to the same super-classwhile we remove $c\%$ of the examples in 4 out of 5 classes. This is performed for both training and validation datasets. Results for ERM and ODD with $p = 10$ and $E = 75$ are shown in Table 2, where ODD is able to outperform ERM in all settings.

4.2   ImageNet

We consider input-agnostic random noise of 0%/20%/40% on the ImageNet-2012 classification dataset [34], where we train ResNet-50 models [10] for 90 epochs and report top-1 and top-5 validation errors in Table 3. ODD significantly outperforms ERM and MentorNet in terms of both top-1 and top-5 errors with 20% and 40% label noise, while being comparable to Luo in these scenarios. Our training time is comparable to that of ERM and much more efficient than Luo since we do not need to differentiate through the Jacobian.

4.3   WebVision

We further verify the effectiveness of our method on a real-world noisy dataset. The WebVision-2017 dataset [21] contains 2.4 million of real-world noisy labels, that are crawled from Google and Flickr using the 1,000 labels from the ImageNet-2012 dataset. We consider training ResNet-50 [38] for 50 epochs, and

Table 4. Top-1 (top-5) accuracy on WebVision and ImageNet validation sets when trained on WebVision.

Method	WebVision	ImageNet
LASS [1]	66.6 (85.6)	59.0 (80.8)
CleanNet [20]	68.5 (86.5)	60.2 (81.1)
ERM	69.7 (87.0)	62.9 (83.6)
MENTORNET [16]	70.8 (88.0)	62.5 (83.0)
CurriculumNet [9]	73.1 (89.2)	64.7 (84.9)
LUO [25]	72.7 (89.5)	64.9 (85.7)
ODD	72.6 (89.3)	64.8 (85.5)

use both WebVision and ImageNet validation sets for 1-crop validation, following the settings in [16]. We do not use a pretrained model or additional labeled data from ImageNet. In Table 4, we demonstrate superior results than most other competitive methods tailored for learning with noisy labels, which suggests that the approach is empirically valid against noisy labels in the real world.

Our ODD method with $p = 30$ removes 9.3% of the total examples. In comparison, we removed around 1.1% of examples in ImageNet; this suggest that WebVision labels are indeed much noisier than the ImageNet labels since there are more examples removed by the (counterfactual) threshold.

4.4   Clothing1M

Table 5. Validation accuracy on Clothing1M.

Method	Setting	Accuracy
ERM	noisy	68.9
GCE	noisy	69.1
Loss Correction [30]	noisy	69.2
LCCN [43]	noisy	71.6
Joint Opt. [39]	noisy	72.2
DMI [42]	noisy	72.5
ODD	noisy	**73.5**
ERM	clean	75.2
Loss Correction	noisy + clean	**80.4**
ODD	noisy + clean	**80.3**

Clothing1M [41] contains 1 million examples with noisy labels and 50,000 examples with clean labels of 14 classes. Following procedures from previous work, we use the ResNet-50 architecture pre-trained on ImageNet, with a starting learning rate of 0.001 trained with 10 epochs. We consider three settings, where

the dataset contains *clean* labels only, *noisy* labels only, or both types of labels. For ODD, we set $E = 1, p = 1$ for the noisy dataset ($E = 1$ because we fine-tune from ImageNet pre-trained model); we then fine-tune on the clean labels if they are available.

Table 5 suggests our method compares favorably against existing methods such as GCE, Joint Optimization [39], latent class-conditional noise model (LCCN, [43]) and Determinant based Mutual Information (DMI, [42]) on the *noisy* dataset, and is comparable to Loss Correction (LC, [30]) on the *noisy + clean dataset*. We note that the complexity of LC scales quadratically in the number of classes, and it would not be feasible for ImageNet or WebVision.

4.5   Ablation Studies

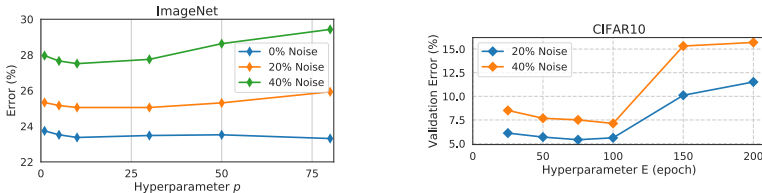We include additional ablation studies in Appendix A.1.

Fig. 7. (Left) ablating p on ImageNet. (Right) ablating E on CIFAR10.

Sensitivity to p We first evaluate noisy ImageNet classification with varying p. A higher p includes more clean examples at the cost of involving more noisy examples. From Fig. 7 (left), ODD is not very sensitive to p in the range of $[0, 30)$, and empirically $p = 10$ represents the best trade-off.

Sensitivity to E We evaluate the validation error of ODD on CIFAR with 20% and 40% input-agnoistic label noise where $E \in \{25, 50, 75, 100, 150, 200\}$ ($E = 200$ is equivalent to ERM). The results in Fig. 7 (right) demonstrate that the effect of E on final performance behaves according to our suggestion, which is to perform ODD when the learning rate is high.

5   Related work

Generalization of SGD Training The generalization of neural networks trained with SGD depend heavily on learning rate schedules [24]. It has been proposed that wide local minima could result in better generalization [13, 3, 17]. Several factors could contribute to wider local optima and better generalization,

such as smaller minibatch sizes [17], reasonable learning rates [18], longer training time [14], or distance from the initialization point [14]. In the presence of mislabeled examples, changes in optimization landscape [1] could result in bad local minima [45], although it is argued that larger batch sizes could mitigate this effect [33].

Training with Mislabeled Examples One paradigm involves estimating the noise distribution [23] or confusion matrix [36]. Another line of methods propose to identify and clean the noisy examples [5] through predictions of auxillary networks [40, 30] or via binary predictions [29]; the noisy labels are either pruned [2] or replaced with model predictions [31]. Our method is comparable to these approaches, but the key difference is that we leverage the implicit regularization of SGD to identify noisy examples. We note that ODD is different from hard example mining [35] which prunes "easier" examples with lower loss; this does not remove mislabeled examples effectively. The method proposed in [29] is most similar to ours in principle, but is restricted to binary classification settings. Other approaches propose to balance the examples via a pretrained network [16], meta learning [32], or surrogate loss functions [7, 47, 39]. Some methods require a set of trusted examples [41, 12].

ODD has several appealing properties compared to existing methods. First, the thresholds for classifying mislabeled examples from ODD do not rely on estimations of the noise confusion matrix. Next, ODD does not require additional trusted examples. Finally, ODD removes potentially noisy examples on-the-fly; it has little computational overhead compared to standard SGD training.

6 Discussion

We have proposed ODD, a straightforward method for robust training with mislabeled examples. ODD utilizes the implicit regularization effect of stochastic gradient descent, which allows us to reason counterfactually about the loss distribution of examples with uniform label noise. Based on quantiles of this (counterfactual) distribution, we can then prune examples that would potentially harm generalization. Empirical results demonstrate that ODD is able to significantly outperform related methods on a wide range of datasets with artificial and real-world mislabeled examples, maintain competitiveness with ERM on clean datasets, as well as detecting mislabeled examples automatically in CIFAR-100.

The implicit regularization of stochastic gradient descent opens up other research directions for implementing robust algorithms. For example, we could consider removing examples not only once but multiple times or retraining from scratch with the denoised dataset. Moreover, it would be interesting to understand the ODD from additional theoretical viewpoints, such as the effects of large learning rates.

References

1. Arpit, D., Jastrzkebski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M.S., Maharaj, T., Fischer, A., Courville, A., Bengio, Y., Lacoste-Julien, S.: A closer look at memorization in deep networks. arXiv preprint arXiv:1706.05394 (June 2017)
2. Brodley, C.E., Friedl, M.A., Others: Identifying and eliminating mislabeled training instances. In: Proceedings of the National Conference on Artificial Intelligence. pp. 799–805 (1996)
3. Chaudhari, P., Choromanska, A., Soatto, S., LeCun, Y., Baldassi, C., Borgs, C., Chayes, J., Sagun, L., Zecchina, R.: Entropy-SGD: Biasing gradient descent into wide valleys. arXiv preprint arXiv:1611.01838 (November 2016)
4. Chaudhari, P., Soatto, S.: Stochastic gradient descent performs variational inference, converges to limit cycles for deep networks. In: 2018 Information Theory and Applications Workshop (ITA). pp. 1–10. IEEE (2018)
5. Cretu, G.F., Stavrou, A., Locasto, M.E., Stolfo, S.J., Keromytis, A.D.: Casting out demons: Sanitizing training data for anomaly sensors. In: Security and Privacy, 2008. SP 2008. IEEE Symposium on. pp. 81–95. IEEE (2008)
6. Du, S.S., Hu, W., Lee, J.D.: Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced. arXiv preprint arXiv:1806.00900 (June 2018)
7. Ghosh, A., Kumar, H., Sastry, P.S.: Robust loss functions under label noise for deep neural networks. In: AAAI. pp. 1919–1925 (2017)
8. Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., He, K.: Accurate, large minibatch SGD: Training ImageNet in 1 hour. arXiv preprint arXiv:1706.02677 (June 2017)
9. Guo, S., Huang, W., Zhang, H., Zhuang, C., Dong, D., Scott, M.R., Huang, D.: Curriculumnet: Weakly supervised learning from large-scale web images. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 135–150 (2018)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385 (December 2015)
11. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing Human-Level performance on ImageNet classification. arXiv preprint arXiv:1502.01852 (February 2015)
12. Hendrycks, D., Mazeika, M., Wilson, D., Gimpel, K.: Using trusted data to train deep networks on labels corrupted by severe noise. arXiv preprint arXiv:1802.05300 (February 2018)
13. Hochreiter, S., Schmidhuber, J.: SIMPLIFYING NEURAL NETS BY DISCOVERING FLAT MINIMA. In: Tesauro, G., Touretzky, D.S., Leen, T.K. (eds.) Advances in Neural Information Processing Systems 7, pp. 529–536. MIT Press (1995)
14. Hoffer, E., Hubara, I., Soudry, D.: Train longer, generalize better: closing the generalization gap in large batch training of neural networks. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30, pp. 1731–1741. Curran Associates, Inc. (2017)
15. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)
16. Jiang, L., Zhou, Z., Leung, T., Li, L.J., Fei-Fei, L.: MentorNet: Learning Data-Driven curriculum for very deep neural networks on corrupted labels. arXiv preprint arXiv:1712.05055 (December 2017)

17. Keskar, N.S., Mudigere, D., Nocedal, J., Smelyanskiy, M., Tang, P.T.P.: On Large-Batch training for deep learning: Generalization gap and sharp minima. arXiv preprint arXiv:1609.04836 (September 2016)
18. Kleinberg, R., Li, Y., Yuan, Y.: An alternative view: When does SGD escape local minima? arXiv preprint arXiv:1802.06175 (February 2018)
19. Krishna, R.A., Hata, K., Chen, S., Kravitz, J., Shamma, D.A., Fei-Fei, L., Bernstein, M.S.: Embracing error to enable rapid crowdsourcing. In: Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems. pp. 3167–3179. CHI '16, ACM, New York, NY, USA (2016). https://doi.org/10.1145/2858036.2858115
20. Lee, K.H., He, X., Zhang, L., Yang, L.: Cleannet: Transfer learning for scalable image classifier training with label noise. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5447–5456 (2018)
21. Li, W., Wang, L., Li, W., Agustsson, E., Van Gool, L.: WebVision database: Visual learning and understanding from web data. arXiv preprint arXiv:1708.02862 (August 2017)
22. Li, Y., Ma, T., Zhang, H.: Algorithmic regularization in over-parameterized matrix sensing and neural networks with quadratic activations. arXiv preprint arXiv:1712.09203 (December 2017)
23. Liu, T., Tao, D.: Classification with noisy labels by importance reweighting. arXiv preprint arXiv:1411.7718 (November 2014)
24. Loshchilov, I., Hutter, F.: SGDR: Stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983 (August 2016)
25. Luo, Y., Zhu, J., Pfister, T.: A simple yet effective baseline for robust deep learning with noisy labels. arXiv preprint arXiv:1909.09338 (2019)
26. Mahajan, D., Girshick, R., Ramanathan, V., He, K., Paluri, M., Li, Y., Bharambe, A., van der Maaten, L.: Exploring the limits of weakly supervised pretraining. arXiv preprint arXiv:1805.00932 (May 2018)
27. Mandt, S., Hoffman, M.D., Blei, D.M.: Stochastic gradient descent as approximate bayesian inference. The Journal of Machine Learning Research **18**(1), 4873–4907 (2017)
28. Neyshabur, B.: Implicit regularization in deep learning. arXiv preprint arXiv:1709.01953 (September 2017)
29. Northcutt, C.G., Wu, T., Chuang, I.L.: Learning with confident examples: Rank pruning for robust classification with noisy labels. arXiv preprint arXiv:1705.01936 (May 2017)
30. Patrini, G., Rozza, A., Menon, A.K., Nock, R., Qu, L.: Making deep neural networks robust to label noise: A loss correction approach. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2233–2241. IEEE (2017)
31. Reed, S., Lee, H., Anguelov, D., Szegedy, C., Erhan, D., Rabinovich, A.: Training deep neural networks on noisy labels with bootstrapping. arXiv preprint arXiv:1412.6596 (December 2014)
32. Ren, M., Zeng, W., Yang, B., Urtasun, R.: Learning to reweight examples for robust deep learning. arXiv preprint arXiv:1803.09050 (March 2018)
33. Rolnick, D., Veit, A., Belongie, S., Shavit, N.: Deep learning is robust to massive label noise. arXiv preprint arXiv:1705.10694 (May 2017)
34. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet large scale visual recognition challenge. International journal of computer vision **115**(3), 211–252 (December 2015). https://doi.org/10.1007/s11263-015-0816-y

35. Shrivastava, A., Gupta, A., Girshick, R.: Training region-based object detectors with online hard example mining. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 761–769 (2016)
36. Sukhbaatar, S., Bruna, J., Paluri, M., Bourdev, L., Fergus, R.: Training convolutional networks with noisy labels. arXiv preprint arXiv:1406.2080 (June 2014)
37. Sun, C., Shrivastava, A., Singh, S., Gupta, A.: Revisiting unreasonable effectiveness of data in deep learning era. arXiv preprint arXiv:1707.02968 (July 2017)
38. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.: Inception-v4, Inception-ResNet and the impact of residual connections on learning. arXiv preprint arXiv:1602.07261 (February 2016)
39. Tanaka, D., Ikami, D., Yamasaki, T., Aizawa, K.: Joint optimization framework for learning with noisy labels. arXiv preprint arXiv:1803.11364 (2018)
40. Veit, A., Alldrin, N., Chechik, G., Krasin, I., Gupta, A., Belongie, S.J.: Learning from noisy Large-Scale datasets with minimal supervision. In: CVPR. pp. 6575–6583 (2017)
41. Xiao, T., Xia, T., Yang, Y., Huang, C., Wang, X.: Learning from massive noisy labeled data for image classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2691–2699 (2015)
42. Xu, Y., Cao, P., Kong, Y., Wang, Y.: L_dmi: An information-theoretic noise-robust loss function. arXiv preprint arXiv:1909.03388 (2019)
43. Yao, J., Wu, H., Zhang, Y., Tsang, I.W., Sun, J.: Safeguarded dynamic label regression for noisy supervision. AAAI (2019)
44. Zagoruyko, S., Komodakis, N.: Wide residual networks. arXiv preprint arXiv:1605.07146 (May 2016)
45. Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O.: Understanding deep learning requires rethinking generalization. arXiv preprint arXiv:1611.03530 (November 2016)
46. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization (October 2017)
47. Zhang, Z., Sabuncu, M.: Generalized cross entropy loss for training deep neural networks with noisy labels. In: Advances in neural information processing systems. pp. 8778–8788 (2018)