Towards Robust Learning with Different Label Noise Distributions

Diego Ortego, Eric Arazo, Paul Albert, Noel E. O'Connor, Kevin McGuinness

Insight Centre for Data Analytics, Dublin City University (DCU)

{diego.ortego, eric.arazo}@insight-centre.org

Abstract—**Noisy labels are an unavoidable consequence of labeling processes and detecting them is an important step towards preventing performance degradations in Convolutional Neural Networks. Discarding noisy labels avoids a harmful memorization, while the associated image content can still be exploited in a semi-supervised learning (SSL) setup. Clean samples are usually identified using the small loss trick, i.e. they exhibit a low loss. However, we show that different noise distributions make the application of this trick less straightforward and propose to continuously relabel all images to reveal a discriminative loss against multiple distributions. SSL is then applied twice, once to improve the clean-noisy detection and again for training the final model. We design an experimental setup based on ImageNet32/64 for better understanding the consequences of representation learning with differing label noise distributions and find that non-uniform out-of-distribution noise better resembles real-world noise and that in most cases intermediate features are not affected by label noise corruption. Experiments in CIFAR-10/100, ImageNet32/64 and WebVision (real-world noise) demonstrate that the proposed label noise Distribution Robust Pseudo-Labeling (DRPL) approach gives substantial improvements over recent state-of-the-art. Code is available at https://git.io/JJ0PV.**

I. Introduction

Modern representation learning, i.e. the extraction of useful information to build classifiers or other predictors [1], in computer vision is led by Convolutional Neural Networks (CNNs) [2]–[8]. Their widespread use is attributable to their capability to model complex patterns when vast amounts of labeled data are available [9]. This supervision requirement limits exploiting the vast amounts of web images as it is infeasible to label them for each particular task. However, leveraging these data might drive visual representation learning a step forward.

What can we do to relax supervision? One could adopt transfer learning or domain adaptation [10], where representations from a source domain are transferred to a target domain where fewer labels are available. This approach, however, makes the target domain dependent on the source one. Learning representations from scratch in the target domain, on the other hand, may lead to better representations. Several alternatives exist: semi-supervised learning (SSL), which jointly learns from few labeled images and extensive unlabeled ones [11], [12]; self-supervised learning, where data provides the supervision [9], [13]; or learning with label noise, where automatic labeling processes introduce noise in the observed labels [14], [15]. This paper focuses on this last alternative, which has attracted much recent interest [15]–[18].

Learning with label noise is challenging; recent studies on the generalization capabilities of deep networks [19] demonstrate that noisy labels are easily fit by CNNs, harming generalization. There is, however, a key observation on how CNNs memorize corruptions: they tend to learn easy patterns first and these patterns are closer to clean data patterns, i.e. correctly labeled images [19], [20], thus exhibiting lower loss than images with noisy or incorrect labels. This phenomenon is commonly named small loss [14], [15], [18] and recent works exploit this *small loss trick* to identify clean and noisy samples [16], [21], [22].

Approaches dealing with label noise can be categorized into: loss correction [16], [23]–[25], when the loss is weighted to correct the label noise effect; relabeling [14], [15], when observed labels are replaced by an estimation of the true labels; and approaches that discard noisy labels to transform the problem into SSL [26], [27]. Despite the variety of approaches and comparative evaluations, it is not clear which of them behave better. Most approaches are exhaustively compared on synthetic label noise in CIFAR data [28] and then tested in real-world datasets such as Clothing1M [29] or WebVision [30]. However, while uniform and non-uniform label flips are considered in CIFAR (in-distribution noise), real-world noise also contains out-of-distribution samples [30], thus leading to a distribution mismatch between synthetic and real data that may impact representation learning. Despite this mismatch, when experimenting with synthetic noise the true labels are known, thus enabling memorization analysis.

In light of these limitations, we undertake an exhaustive study on different label noise distributions and propose a general solution to tackle all of them. In particular, we adopt the 32×32 and 64×64 versions of the ImageNet dataset and artificially introduce 4 different label noise types (2 in-distribution and 2 out-of-distribution). Our analysis reveals differences in the way labels are memorized for different distributions, thus making the application of the small loss trick over training losses harder. We therefore propose a noise detection strategy that reveals a suitable loss to apply the small loss trick regardless of the noise distribution. We find that discarding noisy labels with our method and training in a semi-supervised manner outperforms all other approaches considered. Our contributions include:

- A framework to study label noise with different distributions providing a more complete understanding.
- A label noise detection method agnostic to the noise distribution that substantially outperforms other recent methods [22], [26], [27].

- A study of label noise memorization effect in representation learning, showing that the discrimination power of intermediate representations is not affected.
- An extensive evaluation using multiple label noise distributions and real-world noise, demonstrating both superior performance of our label noise Distribution Robust Pseudo-Labeling (DRPL) approach and contributing to a better understanding of existing methods.

II. RELATED WORK

Label noise is a well-known problem in machine learning [31]. Recent efforts in image classification focus on dealing with in-distribution noise [27], where the set of possible labels S is known and noisy labels belong to this set. However, label noise might also come from outside the distribution [32], which occurs in many real-world scenarios [30].

Several label noise distributions can affect dataset annotations, namely *uniform* or *non-uniform* random label noise. Uniform label noise means the true labels are flipped to a different class with uniform random probability. Non-uniform noise has different flipping probabilities for each class.

Loss correction approaches [16], [18], [24], [25] either modify the loss directly or the network probabilities to compensate for the incorrect guidance provided by the noisy samples. [25] extend the loss with a perceptual term that introduces a reliance on the model prediction. Han et al. [33] extend it by adopting, as a perceptual term, class estimations based on sample prototypes, while Arazo et al. [16] dynamically weight the perceptual term based on the clean-noisy per-sample probability given by a label noise model. Patrini et al. [24] estimate the label noise transition matrix T, which specifies the probability of one label being flipped to another, and correct the softmax probability by multiplying by T. In the same spirit, Yao et al. [18] propose to estimate T in a Bayesian non-parametric form and deduce a dynamic label regression method to train the classifier and model the noise.

Other loss correction approaches reduce the contribution of noisy samples by defining per-sample weights using a mentor network [23], an unsupervised estimation of data complexity [34], or metric learning to pull noisy samples representations away from clean ones [32]. Robust loss functions are studied in [35], which define the generalized cross-entropy loss by jointly exploiting the benefits of mean absolute error and cross-entropy losses. [17] propose the Determinant-based Mutual Information, a generalized version of mutual information that is provably insensitive to noise patterns and amounts.

Other approaches relabel the noisy samples by modeling their noise through conditional random fields [36], or CNNs [37] using a set of clean samples, which limits their applicability. Tanaka et al. [14] and Yi et al. [15] have, however, demonstrated that it is possible to do sample relabeling using the network predictions or estimated label distributions as soft labels.

It has, moreover, recently been demonstrated [26], [27], [38] that it is useful to discard samples that are likely to be noisy while still using them in a semi-supervised setup. [26] define clean samples by inspecting prediction-label agreement, whereas [27] use high softmax probabilities to distinguish clean samples after performing negative learning, i.e. minimizing the probability of predicting likely incorrect classes. More recently, [38] jointly trains two networks and exploits their interactions to dynamically re-define the clean samples during training.

III. FROM LABEL NOISE TO SEMI-SUPERVISED LEARNING

Image classification can be formulated as learning a model $h_\theta(x)$ from a set of training examples $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ with $y_i \in \{0,1\}^C$ being the one-hot encoding true label corresponding to x_i. In our case, h_θ is a CNN and θ represents the model parameters (weights and biases). As we are considering classification under label noise, the label y_i can be noisy (i.e. x_i is a noisy sample). This training under label noise can be redefined through SSL, where the N samples are split into a set of N_u unlabeled samples $\mathcal{D}_u = \{x_i\}_{i=1}^{N_u}$ and a set of N_l labeled samples $\mathcal{D}_l = \{(x_i, y_i)\}_{i=1}^{N_l}$. Ideally, $\mathcal{D}_l$ would contain the clean samples and $\mathcal{D}_u$ the noisy ones. In practice, clean and noisy samples sets are unknown and must be estimated. Detected clean (noisy) samples are used as the labeled (unlabeled) set and, as such, detection accuracy will influence the SSL success. The remainder of this section introduces the proposed two-stage label noise detection (Subsections III-A and III-B) and briefly explains the SSL approach [11] (Subsection III-C).

A. Label noise detection: first stage

Recent literature assumes that using the small loss trick when training with cross-entropy leads to accurate clean-noisy discrimination [16], [21], [22]. However, as Figure 1 shows, different label noise distributions result in different behavior. Noisy samples tend to have higher loss than clean samples for both in-distribution and out-of-distribution noise but the different noise types exhibit different complexities. The uniform noise types (Figure 1 (left part)) exhibit higher separation between the losses of clean (blue) and noisy (red) samples than the non-uniform ones (Figure 1 (right part)). This shows that a straightforward application of the small loss trick will likely encounter some difficulties.

Based on the evidence that clean data is easier to fit than mislabeled data [20], we propose to identify the clean data by fully relabeling all samples using the network predictions and analyzing which samples still fit the original labels. This stage is key for the success of our approach across different label noise distributions. For relabeling we adopt [14], which optimizes the following loss function:

$$\ell_t(\phi) = \begin{cases} -\frac{1}{N} \sum_{i=1}^N y_i^T \log(h_\phi(x_i)) + R & t \le q \\ -\frac{1}{N} \sum_{i=1}^N \tilde{y}_i^T \log(h_\phi(x_i)) + R & t > q \end{cases}, \quad (1)$$

$$R = \lambda_1 R_H + \lambda_2 R_A, \quad (2)$$

where t indexes the epochs, q defines the number of warm-up epochs with the original (potentially noisy) labels y, and R_H and R_A are two regularization terms (see details in [14]) weighted by λ_1 and λ_2 included to ensure convergence. The first phase ($t \le q$) trains without relabeling and uses a high learning rate to prevent fitting the label noise, while the second
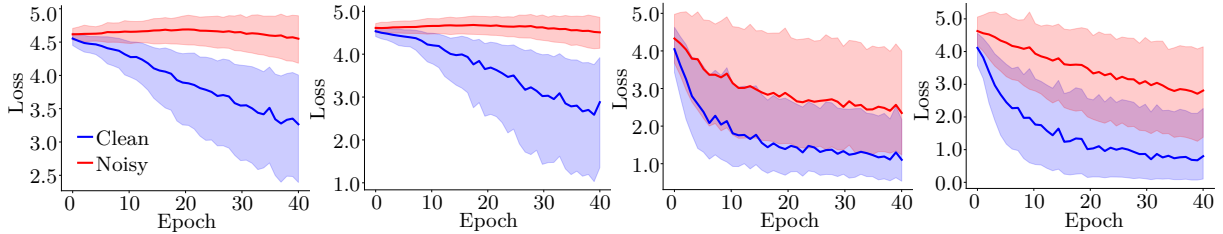
Fig. 1. Loss for clean (blue) and noisy (red) samples for different label noise distributions in 100 classes of ImageNet32. 80% uniform in-distribution noise (left); 80% uniform out-of-distribution noise (mid-left); 50% non-uniform in-distribution noise (mid-right); and 50% non-uniform out-of-distribution noise (right). Training: 40 epochs with a PreAct ResNet-18 [39] with learning rate of 0.1 and cross-entropy loss.

$(t > q)$ relabels all samples by computing soft pseudo-labels $\tilde{y}$ that are re-estimated every epoch using the softmax predictions $h_\phi(x)$. For simplicity we omit index t inside Eq. (1).

In [14], Eq. (1) is used for robust learning against label noise and the pseudo-labels at the end of the training are selected to start a new training stage with improved labels. We conversely, discard noisy labels by exploiting the observation that the new labels or pseudo-labels $\tilde{y}$ no longer represent the original label noise, but the noise from inaccurate network predictions. This alternative view of relabeling results in substantial improvements compared to [14] (see Subsections V-B and V-D). Figure 2 illustrates the benefits of this approach on the detection of clean and noisy samples. The relabeling approach progressively fits the new labels (Figure 2 (left)) as it is highly affected by confirmation bias [11], i.e. over-fitting to incorrect pseudo-labels predicted by the network. However, this reveals an interesting property in the cross-entropy loss $\ell^*(\phi) = -\frac{1}{N} \sum_{i=1}^{N} y_i^T \log(h_\phi(x_i))$ with respect to the original labels y (Figure 2 (mid-left)) that occurs for multiple noise distributions. Pseudo-labels (or predictions) of clean samples tend to agree with the original labels (blue loss is low) substantially better than those of noisy samples (red loss is high). This facilitates distinguishing between clean and noisy samples using the loss, which does not occur in a standard training (Figure 1). Note that the training is guided by Eq. 1 and not by $\ell^*(\phi)$, which is only used for clean/noisy discrimination.

Given the loss $\ell^*(\phi)$ ((*) denotes losses with respect the original labels y), we exploit the small loss trick as in [16], i.e. we fit a 2-component Beta Mixture Model (BMM) to the loss to model clean (noisy) samples using the first (second) component (lower losses correspond to clean samples). The probability of each sample being clean or noisy is then estimated using the posterior probability $p(k \mid \ell)$ from the mixture model. The initial labeled and unlabeled sets are then defined as:

$$\mathcal{D}'_l = \{(x_i, y_i) : p(k = 2 \mid \ell_i^*(\phi)) \leq \gamma_1\}, \tag{3}$$

$$\mathcal{D}'_u = \{(x_i) : p(k = 2 \mid \ell_i^*(\phi)) > \gamma_1\}, \tag{4}$$

where $k = 2$ represents noisy samples, $\ell_i^*(\phi)$ is the loss after the relabeling approach for sample x_i, and γ_1 is a threshold to detect clean and noisy samples. This threshold should be small to select only highly probable clean samples (we use $\gamma_1 = 0.05$ unless otherwise stated).

B. Label noise detection: second stage

The second stage of our clean-noisy samples detector refines the first stage estimation by training a new semi-supervised model $h_\varphi(x)$ using the labeled and unlabeled sets from Eqs. (3) and (4). This new model $h_\varphi(x)$ leads to a loss $\ell^*(\varphi)$ with respect the original labels y that further facilitates the clean-noisy detection as training is performed with far less corrupted labels. Again, we detect clean/noisy samples by fitting a BMM to the loss $\ell^*(\varphi)$ and compute the final labeled and unlabeled sets $\mathcal{D}_l$ and $\mathcal{D}_u$ by selecting samples similarly to Eqs. (3) and (4). As Figure 2 (mid-right) shows, clean and noisy samples become easily separable in the loss $\ell^*(\varphi)$ leading to wider separation in posterior probabilities $p(k \mid \ell^*(\varphi))$ (Figure 2 (right)). This allows the use of a higher threshold compared to γ_1 and lower risk of introducing noisy samples. We therefore apply a maximum a posteriori threshold over $p(k \mid \ell^*(\varphi))$ of $\gamma_2 = 0.5$. Subsection V-A demonstrates the capabilities of our label noise detection method for different noise distributions.

C. The semi-supervised learning approach

We adopt a recent SSL approach described in [11] for simplicity and performance. This approach performs relabeling or pseudo-labeling as presented in the first stage, but the pseudo-labels are only estimated for the unlabeled samples. Mixup data augmentation [40] is also applied to alleviate confirmation bias and make pseudo-labeling effective. The SSL approach is applied twice. The first time SSL is used in the second stage of the label noise detection, i.e. with the labeled $\mathcal{D}'_l$ and unlabeled $\mathcal{D}'_u$ sets (see Subsection III-A) to train $h_\varphi(x)$, whereas the second time it is applied using the final labeled and unlabeled sets $\mathcal{D}_l$ and $\mathcal{D}_u$ (see Subsection III-B) to train the final label-noise robust model $h_\theta(x)$.

IV. EXPERIMENTAL SETUP

CIFAR data [28] is commonly corrupted for fast experimentation with label noise, whereas real-world performance against label noise is evaluated in datasets like WebVision [30]. To the best of our knowledge, however, the related work obviates the differences that might exist between artificially introduced noise and real-world noise. This section introduces the proposed label noise framework using ImageNet32/64 [41] aimed at a better understanding of label noise (Subsection IV-A) and further describes CIFAR-10/100 and WebVision datasets as commonly used frameworks that we also adopt (Subsections IV-B and
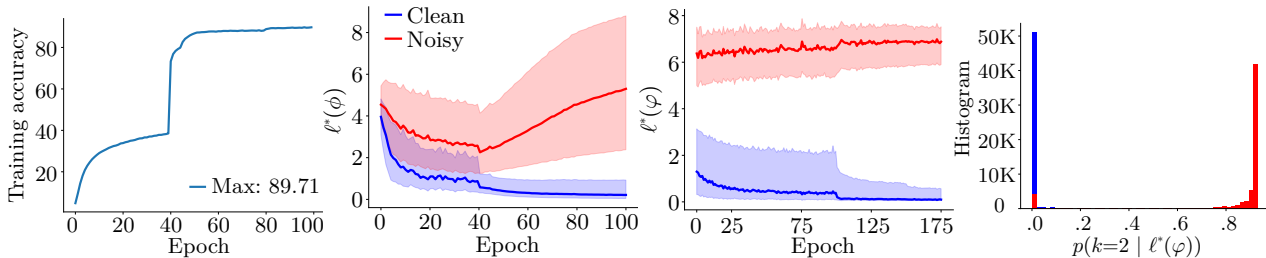
Fig. 2. Proposed approach for noise detection in ImageNet32. Training accuracy for model $h_\phi(x)$ is high due to memorization of pseudo-labels (left). Loss with respect to the original labels $\ell^*(\phi)$ (mid-left) shows that clean samples (blue) still fit the original labels y, while noisy samples (red) do not. Loss $\ell^*(\varphi)$ for the model $h_\varphi(x)$ trained in a semi-supervised manner (mid-right), where clean/noisy samples are further separated. Posterior probability $p(k = 2 \mid \ell^*(\varphi))$ obtained by the BMM (right) used to generate $\mathcal{D}_l$ and $\mathcal{D}_u$. Figures correspond to 50% of non-uniform in-distribution noise (see Subsection II).

IV-C). We use a PreAct ResNet-18 [39] in ImageNet32/64 and CIFAR and a ResNet-18 [42] for WebVision. We always train from scratch (except for the transfer learning experiments in Subsection V-C), using SGD with a momentum of 0.9, a weight decay of 10^{-4}, batch size of 128 and initial learning rate of 0.1 for every stage of our method. Note that we do not use validation sets in any experiment. We take this decision due to the difficulty of defining a clean validation set in a real-world scenario and, more importantly, due to the fact that having clean data allows direct application of SSL, which leads to superior performance [11]. See the supplementary material[1] for a summary of the proposed method and additional results, examples, and configuration details.

A. ImageNet32/64

We propose to use ImageNet32/64 for fast experimentation and higher flexibility in better understanding label noise. ImageNet32/64 are 32×32 and 64×64 downsampled versions of the ImageNet classification dataset [41] (1.2M images uniformly distributed over 1000 classes). To introduce label noise we split the dataset into M in-distribution (ID) classes and $1000 - M$ out-of-distribution (OOD) classes. The split is performed to study both ID noise, as is typically done in the literature [15], and also the less frequently considered OOD noise [32]. We set $M = 100$ randomly selected classes that are always fixed in our experiments, thus leading to 127K images. We study both uniform and non-uniform noise in both ID and OOD scenarios. To introduce uniform noise for ID we randomly flip the true label to another of the M labels using uniform probabilities and excluding the true label, whereas for OOD we randomly select a class among the $1000 - M$ OOD classes and use an image to replace the ID image. To introduce non-uniform noise we use a label noise transition matrix [24] designed to be as realistic as possible. To this end, we average and apply row-wise unit-based normalization to the confusion matrices of the pre-trained ImageNet networks VGG-16 [43], ResNet-50 [42], Inception-v3 [44], and DenseNet-161 [45]. We truncate this 1000×1000 matrix and re-normalize it to the M classes for ID noise and the $1000 - M$ classes for OOD noise. We follow the same process as the uniform case to introduce noise, but using the row distributions corresponding to the

---

[1]https://arxiv.org/abs/1912.08741

---

true label of each image: we randomly flip the label for ID noise, while changing the image content for OOD noise. For a specific noise level r, we always keep $1 - r$ clean samples in each class and modify the remainder.

We use standard data augmentation by random horizontal flips and random 4 (8) pixel translations for ImageNet32 (64) in training. During the first stage of the label noise detection, we train for 40 epochs before starting 60 epochs of relabeling and reduce the learning rate by a factor of 10 in epochs 45 and 80. In the second stage of the label noise detection, we train 175 epochs and reduce the learning rate in epochs 100 and 150. The final SSL stage has 300 epochs with learning rate reductions in epochs 150 and 225.

B. CIFAR-10/100

The CIFAR-10/100 datasets [28] have 10/100 classes of 32×32 images split into 50 (10) K images for training (testing). We follow [15] for label noise addition. For uniform noise, labels are randomly assigned excluding the original label. For non-uniform noise, labels are flipped with probability r to similar classes in CIFAR-10 (i.e. truck $\rightarrow$ automobile, bird $\rightarrow$ airplane, deer $\rightarrow$ horse, cat $\rightarrow$ dog), whereas for CIFAR-100 label flips are done to the next class circularly within the super-classes. Standard data augmentation by random horizontal flips and random 4 pixel translations is used in training. We train as in ImageNet32/64 in CIFAR-100, whereas in CIFAR-10 we slightly modify the first stage by training 130 epochs (70 before relabeling) and reduce the learning rate in epochs 75 and 110. This increase in training epochs is to ensure a better model before relabeling, as classes in CIFAR-10 are more different than in ImageNet32/64 and CIFAR-100, thus incorrect predictions during relabeling are less informative. We use $\gamma_1 = 0.1$ to assure sufficient labeled samples for SSL.

C. WebVision

We use WebVision 1.0 [30] to evaluate performance on real-world label noise. We evaluate our approach using a subset of classes as done in other works [38], [46]. We select all WebVision data for the first 50 classes resulting in a dataset of 137K images. We use random horizontal flips during training and resize images to 256×256 before taking random 224×224 crops. For the first stage of the label noise detection, we train 40 epochs before starting 60 epochs of relabeling and reduce

		NU-ID			U-ID			
		10%	30%	50%	20%	40%	60%	80%
TS [26]	TPR	0.28	0.14	0.12	0.23	0.14	0.15	0.15
	FPR	0.04	0.07	0.09	0.08	0.06	0.07	0.09
NL [27]	TPR	0.80	0.75	0.64	0.83	0.77	0.34	-
	FPR	0.09	0.26	0.34	0.03	**0.03**	**0.00**	-
Ours	TPR	**0.82**	**0.90**	**0.87**	**0.89**	**0.94**	**0.92**	**0.83**
	FPR	**0.02**	**0.04**	**0.11**	**0.02**	0.05	0.05	0.05

		NU-OOD			U-OOD			
		10%	30%	50%	20%	40%	60%	80%
TS [26]	TPR	0.40	0.36	0.27	0.38	0.32	0.30	0.29
	FPR	0.04	**0.06**	**0.06**	0.03	0.02	0.02	**0.06**
NL [27]	TPR	0.82	0.81	0.75	0.80	0.79	0.66	-
	FPR	0.10	0.18	0.22	**0.03**	0.03	**0.01**	-
Ours	TPR	**0.83**	**0.86**	**0.85**	**0.89**	**0.90**	**0.87**	**0.79**
	FPR	0.07	0.23	0.32	0.04	0.06	0.06	**0.06**

the learning rate dividing by 10 twice (epochs 45 and 80). For the second stage, we train 150 epochs and reduce the learning rate in epochs 100 and 125. The final SSL stage has 200 epochs with learning rate reductions in epochs 150 and 175.

V. EXPERIMENTS

A. Label noise detection comparison

Transforming the supervised training with label noise into SSL requires detecting the noise to, ideally, discard the labels and turn noisy samples into unlabeled ones. As commented in Section I, many approaches use the small loss trick, i.e. considering low loss samples as clean ones, to accomplish such detection. However, Figure 1 shows that different noise distributions present different challenges, limiting a straightforward application of this trick. We confirm this limitation in Figure 3, where we compare, for different label noise distributions, the Receiver Operating Characteristic (ROC) curves for our label noise detection method and the small loss trick (using cross-entropy with and without mixup [40]). A straightforward application of the small loss trick [16], [21], [22] (C and M losses) is relatively robust for uniform noises (left and mid-right), but poorly performs for non-uniform ones (mid-left and right). Conversely, the proposed approach robustly detects noisy samples for different noise distributions, thus supporting the effectiveness of our method. We also outperform two recently proposed label noise detection methods [26], [27] across different label noise distributions (see Table I). Note that we encounter some limitations addressing high levels of non-uniform OOD noise, which occurs due to the nature of the classes used as noise. We are using the ImageNet confusion matrix in the validation set to introduce noise and we have 100 (900) in- (out-of-) distribution classes, thus using the most challenging classes as OOD noise.

B. Comparison with related work

We select representative top-performing loss correction [16], [24], relabeling [14], and label noise robust regularization approaches [40] to compare against our label noise Distribution Robust Pseudo-Labeling (DRPL) approach in Table II. The proposed method gives remarkable improvements across different levels and distributions of label noise. Note that, unlike most methods compared, our method shows little degradation between the best (reported in Table II) and last epoch accuracy (see extended results in the supplementary material). In general, R [14] behaves consistently across label noise levels and distributions, while DB [16] has problems with non-uniform noise. FW [24] and M [40] tend to exhibit worse performance. An important observation is that non-uniform OOD noise exhibits less degradation than other noise types for all methods. This is reasonable as OOD samples whose content is close to an ID class will contribute to improved representation learning due to semantic similarities, and the network predictions for these samples will not harm the model for ID classes. This behavior resembles real-world noisy data as observed in the WebVision dataset results in Subsection V-D.

C. Effects of label noise memorization

We know that clean (easier) patterns are learned first by deep neural networks and that noisy labels can be completely memorized [19], [20]. *How do networks memorize noisy samples?* Figure 4 provides some intuition. It shows the Class Activation Maps [47] for the true class and the predicted class for undetected noisy samples. The network skips relevant areas for the true class, attending to areas that might help explain (i.e. memorizing) the noisy label. For example, the network extends the class activation maps to cover larger regions (columns 2, 4) such as the whole lamp instead of the candles for the noisy label *lobster* (column 2), while other times it also omits characteristic areas of the true class (columns 1, 3 and 5) such as the right part of the bridge to better fit the noisy label *dock* (column 1).

Does memorization negatively impact visual representation learning? We follow the standard approach of using linear probes [48] to verify the utility of features under different noise levels and distributions in a target task. Specifically, we train a linear classifier on the global average pooled activations obtained after each of the 4 PreAct ResNet-18 blocks. Figure 5 presents the results in ImageNet64. Our model (O) clearly outperforms mixup (M) in the target task performance using features from the last block. Better source models (i.e. trained with less noise) also tend to produce better target performance. However, an interesting finding is that for both uniform and non-uniform noise, the final accuracy exhibits degradation in L4, while for earlier features no degradation is observed even for M (a model that has memorized the noise). An exception is 80% of uniform noise, where degradation is found across all blocks, but with more discriminative representations learned by our approach. Similar results are observed in ImageNet32 (see supplementary material).

D. Other datasets and real-world frameworks

Table III compares DRPL against related work in the first 50 classes of WebVision 1.0 dataset [30] to verify the
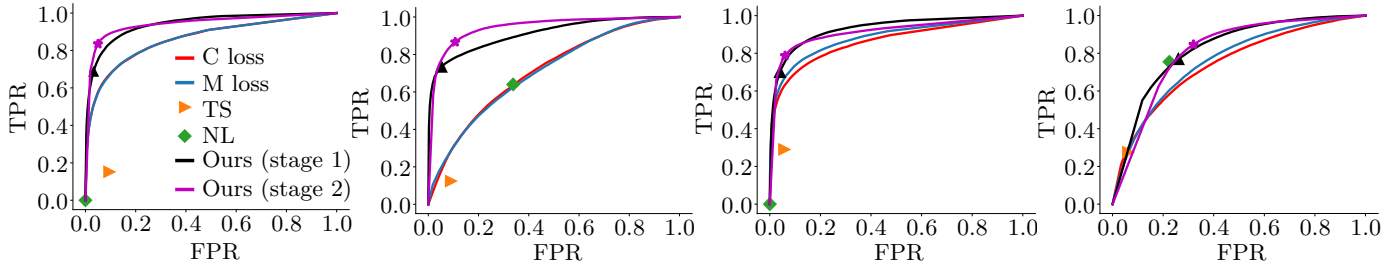
Fig. 3. Label noise detection methods in ImageNet32. 80% uniform ID (left) and OOD (mid-left) noise. 50% non-uniform ID (mid-right) and OOD (right) noise. The proposed method (star in stage 2 denotes operating point) surpasses detection based on cross-entropy loss without (C) and with mixup (M), TS [26] and NL [27]. Key: ID (in-distribution). OOD (out-of-distribution).

TABLE II
IMAGENET32/64 ACCURACY. BOLD DENOTES BEST PERFORMANCE. KEY: NU (NON-UNIFORM); U (UNIFORM); ID (IN-DISTRIBUTION); OOD (OUT-OF-DISTRIBUTION).

| | ImageNet32 | | | | | | | | ImageNet64 | | | | | | | |
| | NU-ID | | U-ID | | NU-OOD | | U-OOD | | NU-ID | | U-ID | | NU-OOD | | U-OOD | |
	30%	50%	40%	80%	30%	50%	40%	80%	30%	50%	40%	80%	30%	50%	40%	80%
FW [24]	54.22	43.38	52.06	31.20	62.14	55.06	56.32	40.08	60.10	46.06	57.42	37.84	69.86	63.38	63.08	47.68
R [14]	67.24	63.62	62.98	41.52	66.36	62.80	64.04	45.00	74.28	69.20	70.98	48.44	74.22	70.74	72.78	54.00
M [40]	67.14	51.96	61.98	38.92	66.14	60.62	64.66	47.40	74.02	58.14	69.90	49.22	74.78	69.40	73.94	59.54
DB [16]	62.88	52.20	67.62	45.34	64.86	60.58	65.96	39.30	71.30	60.98	74.56	56.44	77.94	70.38	74.08	50.98
DRPL	**73.46**	**68.18**	**73.48**	**61.78**	**71.38**	**67.32**	**71.36**	**54.10**	**81.90**	**77.66**	**81.50**	**73.08**	**80.44**	**76.38**	**79.76**	**64.34**

TABLE III
WEBVISION (50 CLASSES) ACCURACY. KEY: NU (NON-UNIFORM NOISE). U (UNIFORM NOISE). CE (CROSS-ENTROPY).

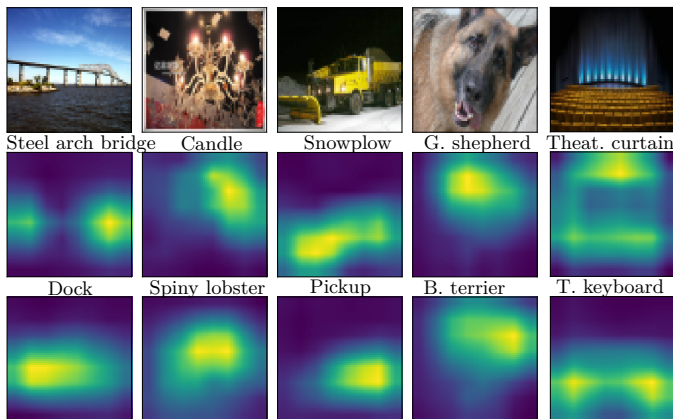| CE	FW [24]	R [14]	M [40]	GCE [35]	DB [16]	DMI [17]	P [15]	DM [38]	DRPL
73.88	74.68	76.52	80.76	74.28	79.68	73.96	79.96	78.16	**82.08**

Fig. 4. Label noise memorization for undetected noisy images in ImageNet64 (50%, non-uniform in-distribution noise). From top to bottom: image and Class Activation Map (CAM) for the true and predicted class. Note that the predicted class (bottom) is also the noisy label used during training.

practical use of our method in real-world noise scenarios. The proposed approach is more accurate than all compared approaches. Note, however, that many approaches give very similar performance. Surprisingly, a straightforward approach like M [40], not specifically designed to deal with label noise, gives the second highest accuracy. The similar performance among approaches is something that can be also observed for

TABLE IV
CIFAR-10(100) ACCURACY ON TOP (BOTTOM). KEY: NU (NON-UNIFORM NOISE). U (UNIFORM NOISE). CE (CROSS-ENTROPY).

| | 0% | NU | | | U | | | |
		10%	30%	40%	20%	40%	60%	80%
CE	93.87	90.97	90.22	88.16	87.75	83.32	75.71	43.69
FW [24]	94.61	90.85	87.95	84.85	85.46	80.67	70.86	45.58
R [14]	94.37	93.70	92.69	92.70	92.77	89.97	85.62	53.26
M [40]	**96.07**	93.79	91.38	87.01	91.27	85.84	80.78	57.93
GCE [35]	93.93	91.40	90.45	88.39	88.49	84.09	76.55	43.39
DB [16]	92.78	91.77	93.23	91.25	93.95	92.38	**89.53**	49.90
DMI [17]	93.93	91.31	91.34	88.64	88.40	83.98	75.91	44.17
PEN [15]	93.94	93.19	92.94	91.63	92.87	91.34	89.15	56.14
DRPL	94.47	**95.70**	**93.65**	**93.14**	**94.20**	**92.92**	89.21	**64.35**

| | 0% | NU | | | U | | | |
		10%	30%	40%	20%	40%	60%	80%
CE	74.59	68.18	54.20	46.55	59.19	51.44	39.05	19.59
FW [24]	75.43	68.82	54.65	45.38	61.03	51.44	39.05	19.59
R [14]	74.43	73.09	68.25	59.49	70.79	66.35	57.48	30.65
M [40]	**78.33**	73.39	59.15	49.40	66.60	54.69	45.80	27.02
GCE [35]	74.91	68.34	55.56	47.24	60.09	61.23	49.75	25.77
DB [16]	70.64	68.19	62.81	55.76	69.12	64.84	57.85	46.45
DMI [17]	74.75	68.29	54.40	46.65	59.16	53.49	41.49	20.50
PEN [15]	77.80	**76.31**	63.67	50.64	**75.16**	69.56	56.16	27.12
DRPL	72.27	72.40	**69.30**	**65.86**	71.25	**73.13**	**68.71**	**53.04**

non-uniform OOD noise in ImageNet32/64. Another similarity here with non-uniform OOD noise is the small degradation
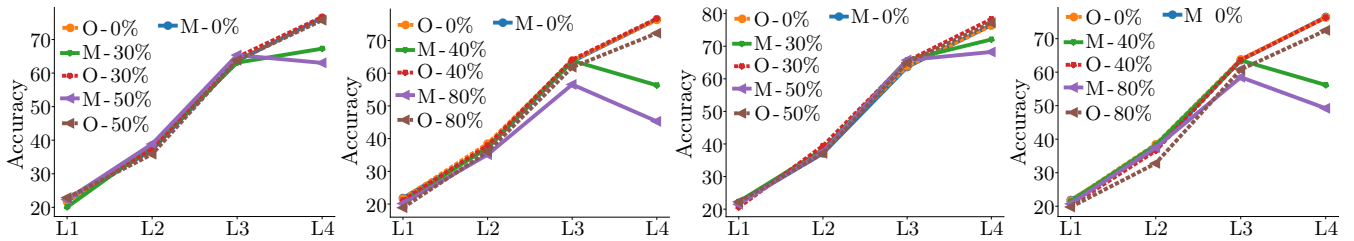
Fig. 5. ImageNet64 linear probes. A linear classifier is trained using features at different depths. Source task: 100 classes. Target task: 25 classes from the remaining 900. Noise: in-distribution non-uniform (left) and uniform (mid-left), and out-of-distribution non-uniform (mid-right) and uniform (right). Model from the last training epoch in the source domain is used (i.e. M has fitted the noise). Key: M: Mixup. O: Ours.

Fig. 6. Examples of detected noisy images in WebVision [30]. The first column shows a clean example from the class.

in performance at the last training epoch (see supplementary material). Furthermore, the reduced noise level in WebVision, estimated at around 20% in [30], and with 17% of detected noisy samples in our experiment, helps to explain performance similarities across approaches: this is also seen in CIFAR and ImageNet32/64 with low noise levels. We present detected noisy images in Figure 6 (more examples in the supplementary material), which are mainly from outside the distribution.

We also evaluate DRPL in CIFAR-10/100 (see Table IV), showing improved performance over all other compared approaches (extended results in the supplementary material). We evaluate the same approaches as in ImageNet32/64 [14], [16], [24], [40] and also add further recent approaches [15], [17], [35] (configuration details in supplementary material). Improved results for uniform and non-uniform noise in CIFAR confirm the improvements observed in ImageNet32/64. The additional recent approaches, GCE [35], DMI [17], and PEN [15] are outperformed by our approach DRPL in most cases with GCE and DMI far from top-performance. The exception is CIFAR-100 with little or no noise where top performances of 78.33 for M [40] and 76.31 for PEN [15] surpasses DRPL, thus suggesting that the information discarded by our approach is important for achieving top performance. This highlights that high loss is indicative not only of incorrect labels, but also of difficult samples [49] and it is the semi-supervised approach that determines whether this information is recovered.

Note that it is unfair to directly compare against numbers reported in DM [38] as they use a clean set to select different hyperparameters for each noise-level and type. We prefer not to use a clean set and adopt a common configuration, which

makes comparisons fairer. However, Table III reports our run of DM (official code and configuration) in WebVision, where there are no different noise types and levels, showing that the proposed DRPL approach outperforms DM by 4 accuracy points. DM [38] also reports 77.32 accuracy results in a different WebVision setup (subset of the first 50 classes and InceptionResNetv2 network). Our approach under that setup achieves 77.12, demonstrating that the proposed approach is, at very least, competitive with the latest state-of-the-art.

It is worth noting that methods behave differently against artificially introduced noise in CIFAR than in real-world scenarios like WebVision [30]. Substantial accuracy degradation is seen in many methods in CIFAR at the end of the training (see supplementary material), while in WebVision degradations are minor, and mixup (M) [40], which does not deal specifically with label noise, outperforms most approaches. We believe that evaluation with OOD noise in ImageNet32/64 helps in understanding how label noise memorization might behave in real-world scenarios.

VI. Conclusion

We propose a framework to study multiple label noise distributions and a straightforward approach based on label noise detection and semi-supervised learning to tackle them all. We provide intuitions about the network behavior when memorizing noisy labels and show that such memorization does not often harm learning discriminative intermediate representations. Results in five datasets support the generality and robustness of our approach and help in reducing the gap between synthetic and real-world label noise. Future work will study more noise distributions and their combinations to further approach real-world noise.

Acknowledgement

References

[1] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.
[2] W. Beluch, T. Genewein, A. Nürnberger, and J. Köhler, "The Power of Ensembles for Active Learning in Image Classification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[3] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Deep image homography estimation," *arXiv:1606.03798*, 2016.

[4] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional Two-stream Network Fusion for Video Action Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[5] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. C. Niebles, "Dense-Captioning Events in Videos," in *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[6] Y. Ono, E. Trulls, P. Fua, and K. Moo Yi, "LF-Net: Learning Local Features from Images," *arXiv: 1805.09662*, 2018.

[7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[8] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid Scene Parsing Network," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[9] A. Kolesnikov, X. Zhai, and L. Beyer, "Revisiting Self-Supervised Visual Representation Learning," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[10] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, 2018.

[11] E. Arazo, D. Ortego, P. Albert, N. O'Connor, and K. McGuinness, "Pseudo-Labeling and Confirmation Bias in Deep Semi-Supervised Learning," *arXiv: 1908.02983*, 2019.

[12] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. Raffel, "MixMatch: A Holistic Approach to Semi-Supervised Learning," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[13] D. Pathak, R. Girshick, P. Dollár, T. Darrell, and B. Hariharan, "Learning Features by Watching Objects Move," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[14] D. Tanaka, D. Ikami, T. Yamasaki, and K. Aizawa, "Joint Optimization Framework for Learning with Noisy Labels," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[15] K. Yi and J. Wu, "Probabilistic End-To-End Noise Correction for Learning With Noisy Labels," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[16] E. Arazo, D. Ortego, P. Albert, N. O'Connor, and K. McGuinness, "Unsupervised Label Noise Modeling and Loss Correction," in *International Conference on Machine Learning (ICML)*, 2019.

[17] Y. Xu, P. Cao, Y. Kong, and Y. Wang, "L_DMI: An Information-theoretic Noise-robust Loss Function," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[18] J. Yao, H. Wu, Y. Zhang, I. Tsang, and J. Sun, "Safeguarded Dynamic Label Regression for Noisy Supervision," in *Association for the Advancement of Artificial Intelligence Conference on Artificial Intelligence (AAAI)*, 2019.

[19] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires re-thinking generalization," in *International Conference on Learning Representations (ICLR)*, 2017.

[20] D. Arpit, S. Jastrzebski, N. Ballas, D. Krueger, E. Bengio, M. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio, and S. Lacoste-Julien, "A Closer Look at Memorization in Deep Networks," in *International Conference on Machine Learning (ICML)*, 2017.

[21] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[22] H. Song, M. Kim, and J.-G. Lee, "SELFIE: Refurbishing Unclean Samples for Robust Deep Learning," in *International Conference on Machine Learning (ICML)*, 2019.

[23] L. Jiang, Z. Zhou, T. Leung, L. Li, and L. Fei-Fei, "MentorNet: Learning Data-Driven Curriculum for Very Deep Neural Networks on Corrupted Labels," in *International Conference on Machine Learning (ICML)*, 2018.

[24] G. Patrini, A. Rozza, A. Krishna Menon, R. Nock, and L. Qu, "Making Deep Neural Networks Robust to Label Noise: A Loss Correction Approach," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[25] S. Reed, H. Lee, D. Anguelov, C. Szegedy, D. Erhan, and A. Rabinovich, "Training deep neural networks on noisy labels with bootstrapping," in *International Conference on Learning Representations (ICLR)*, 2015.

[26] Y. Ding, L. Wang, D. Fan, and B. Gong, "A Semi-Supervised Two-Stage Approach to Learning from Noisy Labels," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2018.

[27] Y. Kim, J. Yim, J. Yun, and J. Kim, "NLNL: Negative Learning for Noisy Labels," in *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[28] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," University of Toronto, Tech. Rep., 2009.

[29] T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang, "Learning from massive noisy labeled data for image classification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[30] W. Li, L. Wang, W. Li, E. Agustsson, and L. Van Gool, "WebVision Database: Visual Learning and Understanding from Web Data," *arXiv: 1708.02862*, 2017.

[31] B. Frenay and M. Verleysen, "Classification in the Presence of Label Noise: A Survey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 5, pp. 845–869, 2014.

[32] Y. Wang, W. Liu, X. Ma, J. Bailey, H. Zha, L. Song, and S.-T. Xia, "Iterative Learning With Open-Set Noisy Labels," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[33] J. Han, P. Luo, and X. Wang, "Deep Self-Learning From Noisy Labels," in *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[34] S. Guo, W. Huang, H. Zhang, C. Zhuang, D. Dong, M. Scott, and D. Huang, "CurriculumNet: Weakly Supervised Learning from Large-Scale Web Images," in *European Conference on Computer Vision (ECCV)*, 2018.

[35] Z. Zhang and M. Sabuncu, "Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[36] A. Vahdat, "Toward Robustness against Label Noise in Training Deep Discriminative Neural Networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[37] A. Veit, N. Alldrin, G. Chechik, I. Krasin, A. Gupta, and S. Belongie, "Learning From Noisy Large-Scale Datasets With Minimal Supervision," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[38] J. Li, R. Socher, and S. Hoi, "DivideMix: Learning with Noisy Labels as Semi-supervised Learning," in *International Conference on Learning Representations (ICLR)*, 2020.

[39] K. He, X. Zhang, S. Ren, and J. Sun, "Identity Mappings in Deep Residual Networks," in *European Conference on Computer Vision (ECCV)*, 2016.

[40] H. Zhang, M. Cisse, Y. Dauphin, and D. Lopez-Paz, "mixup: Beyond Empirical Risk Minimization," in *International Conference on Learning Representations (ICLR)*, 2018.

[41] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[43] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv: 1409.1556*, 2014.

[44] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[45] G. Huang, Z. Liu, L. Van der Maaten, and K. Weinberger, "Densely Connected Convolutional Networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[46] P. Chen, B. Liao, G. Chen, and S. Zhang, "Understanding and Utilizing Deep Neural Networks Trained with Noisy Labels," in *International Conference on Machine Learning (ICML)*, 2019.

[47] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning Deep Features for Discriminative Localization," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[48] R. Zhang, P. Isola, and A. Efros, "Split-Brain Autoencoders: Unsupervised Learning by Cross-Channel Prediction," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[49] A. Jiang, D.-K. Wong, G. Zhou, D. Andersen, J. Dean, G. Ganger, G. Joshi, M. Kaminksy, M. Kozuch, Z. Lipton, and P. Pillai, "Accelerating Deep Learning by Focusing on the Biggest Losers," *arXiv: 1910.00762*, 2019.