Centrality and Consistency: Two-Stage Clean Samples Identification for Learning with Instance-Dependent Noisy Labels

Ganlong Zhao[1,2][0000−0002−1612−3641], Guanbin Li[1⋆][0000−0002−4805−0926], Yipeng Qin[3][0000−0002−1551−9126], Feng Liu[4][0000−0002−4811−7828], and Yizhou Yu[2⋆][0000−0002−0470−5548]

[1] Sun Yat-sen University, Guangzhou 510006, China
[2] The University of Hong Kong, Hong Kong, China
[3] Cardiff University, Cardiff, United Kingdom
[4] Deepwise AI Lab, Beijing, China
zhaogl@connect.hku.hk, liguanbin@mail.sysu.edu.cn, QinY16@cardiff.ac.uk, liufeng@deepwise.com, yizhouy@acm.org

Abstract. Deep models trained with noisy labels are prone to overfitting and struggle in generalization. Most existing solutions are based on an ideal assumption that the label noise is class-conditional, *i.e.* instances of the same class share the same noise model, and are independent of features. While in practice, the real-world noise patterns are usually more fine-grained as instance-dependent ones, which poses a big challenge, especially in the presence of inter-class imbalance. In this paper, we propose a two-stage clean samples identification method to address the aforementioned challenge. First, we employ a class-level feature clustering procedure for the early identification of clean samples that are near the class-wise prediction centers. Notably, we address the class imbalance problem by aggregating rare classes according to their prediction entropy. Second, for the remaining clean samples that are close to the ground truth class boundary (usually mixed with the samples with instance-dependent noises), we propose a novel consistency-based classification method that identifies them using the consistency of two classifier heads: the higher the consistency, the larger the probability that a sample is clean. Extensive experiments on several challenging benchmarks demonstrate the superior performance of our method against the state-of-the-art. Code is available at `https://github.com/uitrbn/TSCSI_IDN`.

Keywords: Instance-Dependent Noise, Noisy Label, Image Classification.

1 Introduction

Deep learning has shown transformative power in various real-world applications but is notoriously data-hungry[10,11,29,9,21,45]. There are some other alternatives which try to reduce the cost of human labor for data annotation, such as

⋆ Corresponding authors are Guanbin Li and Yizhou Yu.

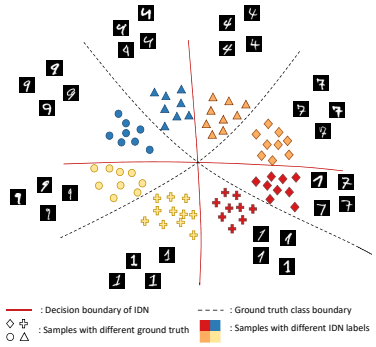


Fig. 1: Example of IDN. The different shapes of the markers represent different ground truth classes. The different colors of the markers represent the noisy (IDN) labels. Different from random noise, IDN samples tend to be distributed near the ground truth class boundary, thus confusing the classifier and leading to over-fitted decision boundaries.

crawling web images and using machine-generated labels. However, such data are usually noisy, which impedes the generalization of deep learning models due to over-fitting.

Addressing the aforementioned issue, Learning with Noisy Labels (LNL) was proposed as a new topic and has attracted increasing attention in both academia and industry. Existing LNL methods mostly focus on the learning with class-conditional noise (CCN), which aims to recover a noise transition matrix that contains class-dependent probabilities of a clean label flipping into a noisy label. However, CCN is too ideal for real-world LNL as it ignores the dependence of noise on the content of individual images, *a.k.a.* instance-dependent noise (IDN).

Unlike random noise or CCN that can be countered by collecting more (noisy) data[4], IDN has some important characteristic that makes it difficult to be tackled. First, classifiers can easily over-fit to the IDN because the noisy labels are dependent on sample features. As Fig. 1 shows, mislabeled IDN samples (samples with the same shape but with different colors) share similar image features to their mislabeled classes, and thus tend to be distributed near the boundary between their ground truth class and the mislabeled class. As a result, the classifier can easily be confused and over-fits to IDN samples, leading to specious decision boundaries (red lines in Fig. 1). In addition, the challenge of IDN can be further amplified in the presence of inter-class imbalance and differences. Consider Clothing1M [38], an IDN dataset verified by [3], in which the noise is highly imbalanced and asymmetric. In Clothing1M, the IDN samples are unevenly distributed as the samples from similar classes (*e.g.* sweater and knitwear) can be extremely ambiguous, while those from other classes (*e.g.* shawl and underwear) are easily distinguishable. Such unevenly distributed IDN samples can be further amplified by the class imbalance problem, as there is no guarantee of a balanced dataset due to the absence of ground truth labels.

Fig. 2: The transition matrix of Clothing1M copied from [38]. The distribution of noisy labels are highly imbalanced. Some classes are almost clean (e.g. Shawl) while some classes has more mislabeled samples than correct labels (e.g. Sweater).

In this paper, we follow DivideMix [17] that formulates LNL as a semi-supervised learning problem and propose a novel two-stage method to identify clean versus noisy samples in the presence of IDN and the class imbalance problem. In the first stage, we employ a class-level feature-based clustering procedure to identify easily distinguishable clean samples according to their cosine similarity to the corresponding class-wise prediction centers. Specifically, we collect the normalized features of samples belonging to different classes respectively and calculate their class-wise centers located on a unit sphere. Then, we apply Gaussian Mixture Model (GMM) to binarily classify the samples according to their cosine similarity to their corresponding class centers and identify the ones closer to class centers as clean samples. Notably, we propose to augment the GMM classification by aggregating rare classes based on their prediction entropy, thereby alleviating the impact of the class imbalance problem. In the second stage, we propose a consistency-based classification method to identify the hard clean samples that are mixed with IDN samples around the ground truth class boundaries. Our key insight is that such clean samples can be identified by the prediction consistency of two classifiers. Compared to IDN samples, clean samples should produce more consistent predictions. Specifically, we incorporate two regularizers into the training: one applied to the feature extractor to encourage it to facilitate consistent outputs of the two classifiers; one applied to the two classifiers to enforce them generating inconsistent predictions. After training, we use another GMM to binarily classify the samples with smaller GMM means as clean samples. After identifying all clean samples, we feed them into the semi-supervised training as labeled samples, thereby implementing our learning with instance-dependent noisy labels. In summary, our contributions could be summarized as:

– We propose a method that delving into the instance-dependent noise, and design a class-level feature clustering procedure focusing on the imbalanced and IDN samples detection.

- We further propose to identify the hard clean samples around the ground truth class boundaries by measuring the prediction consistency between two in-dependently trained classifiers, and further improves the accuracy of clean versus noisy classification.
- Our method achieves state-of-the-art performance in some challenging benchmarks, and is proved to be effective in different kinds of synthetic IDN.

2 Related Work

A large proportion of previous LNL methods focus on the class-conditional noise. With the class-conditional noise assumption, some methods try to correct the loss function with the noise transition matrix[27], which can be estimated through exploiting a noisy dataset[19,27,35,47] or using a clean set of data[12,44]. Such loss correction methods based on noise transition matrix is infeasible for instance-dependent noise, since the matrix is dataset dependent and the number of parameters grows proportionally with the size of training dataset.

Some methods seek to correct the loss by reweighting the noisy samples or selecting the clean data [33,15]. A common solution is to treat the samples with smaller loss as clean data[17,31,13]. However, as pointed out by [3], instance-dependent noise can be more easily over-fitted, and the memorization effect, which indicates that CNN-based models always tend to learn the general simple pattern before over-fitting to the noisy labels, becomes less significant when the model is trained with instance-dependent noise.

Some other methods combat the noisy label with other techniques. For example, Kim *et al*. [14] combine positive learning with negative learning, which uses the complementary labels of noisy data for model training. Some methods[17,25] formulate LNL as a semi-supervised learning problem. DivideMix[17] divides the dataset into clean and noisy sets, which serve as labeled and unlabeled data for semi-supervised learning. Some methods investigate the influence of augmentation strategy[26] or enforce the prediction consistency between different augmentations[22]. C2D[43] utilizes self-supervised learning to facilitate the learning with noisy labels.

Chen *et al*. [5] pointed out that for diagonally-dominant class-conditional noise, one can always obtain an approximately optimal classifier by training with a sufficient number of noisy samples. And it raise the significance of learning with IDN. There has been some works for this topic. CORES^2[5] try to progressively sieve out corrupted samples and avoid specifying noise rate. CAL[46] propose a second-order approach with the assistance of additional second-order statistics. Besides, some research work also propose methods for IDN generation[3,36].

3 Method

3.1 Overview

The classification of noisy versus clean samples by the model outputs and their labels is a prevalent choice in the learning with noisy labels (LNL). Previous

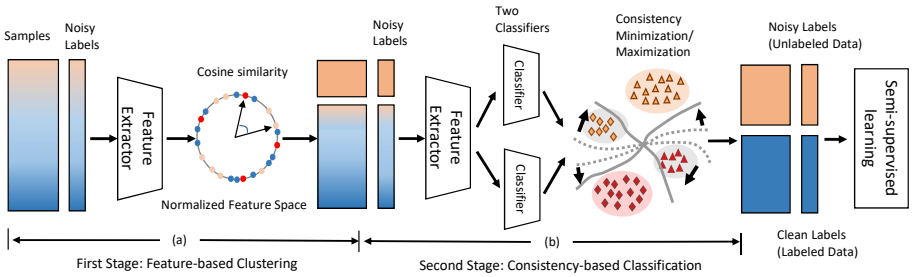


Fig. 3: The overview of our proposed method. (a) The first stage. The noisy samples and labels are sent to the feature extractor for calculating the normalized features. The features are clustered with the prediction of samples. Noisy samples are divide to clean set and noisy set according to the cosine similarity between the feature and the center of its labels. (b) The model is train to minimize/maximize the prediction between two classifier heads and samples with smaller consistency are identified as noisy labels. (c) The clean/noisy set serve as labeled/unlabeled data for semi-supervised training.

studies use the cross-entropy of noisy samples [17] or confidence thresholds [40] for noisy versus clean division. However, as Chen *et al.* [3] point out, samples with instance-dependent noise (IDN) can be more easily over-fitted by neural networks, resulting in less reliable model outputs that confuse the classification of clean versus noisy samples. Such confusion is further amplified when the noisy dataset is imbalanced. For example, the differences between clean and noisy samples might be neglected for rare classes that contribute little to the overall prediction accuracy.

Therefore, we propose a two-stage method which can effectively address IDN in the presence of class imbalance. In the first stage, we leverage a class-level feature-based clustering process to identify easily distinguishable clean samples that are close to their corresponding class centers in the feature space. Specifically, in this stage, we address the class imbalance by aggregating rare classes identified by their prediction entropy. In the second stage, we address the remaining clean samples, which are close to the ground truth class boundaries and are thus mixed with IDN samples. Our key insight is that such clean samples can be identified by the consistent predictions of two classifiers. Specifically, we propose a mini-max strategy for this consistency-based clean versus noisy classification: we simultaneously regularize the two classifiers to generate inconsistent predictions but enforce the feature extractor to facilitate the two classifiers to generate consistent predictions. After training, we identify the clean samples as the ones that lead to more consistent predictions between the two classifiers. After identifying all clean samples, we follow DivideMix [17] and implement the learning with instance-dependent noisy labels as a semi-supervised learning problem that takes the clean samples as labeled samples, and the rest (noisy) samples as unlabeled samples.

3.2 Feature-based Clustering

As common practice, we divide a CNN-based classifier into two parts: a feature extractor F that takes images as input and extracts their features, and the following classifier G that outputs classification probabilities based on the image features extracted by F. Given a noisy dataset $\{x_i, \bar{y}_i\}_{i=1}^{N}$, where x_i is an image sample and $\bar{y}_i$ is its (noisy) label, we denote $\hat{f}_i = \frac{f_i}{\|f_i\|}$ as the normalized feature of x_i extracted by F, *i.e.* $f_i = F(x_i)$, $\hat{y}_i = G(f_i)$ as the predicted label of x_i, and calculate the class-wise feature centers O_c according to $\hat{y}_i$ as:

$$O_c = \frac{\sum_{i=1}^{N_c} \hat{f}_i}{\| \sum_{i=1}^{N_c} \hat{f}_i \|}, \tag{1}$$

where $c \in \{1, 2, 3, ..., C\}$ denotes the C classes, N_c is the number of samples x_i whose noisy label $\bar{y}_i = c$. Then, we can obtain the cosine similarity between each sample x_i and its corresponding feature center $O_{\bar{y}_i}$ as:

$$S_i = \hat{f}_i \cdot O_{\bar{y}_i}. \tag{2}$$

Finally, we apply class-wise Gaussian Mixture Model (GMM) to the similarities S_i of samples for each class and performs binary classification. As the cosine similarity of noisy samples tend to be smaller, the component of GMM with a larger mean, *i.e.* larger similarity, is denoted as the clean set. Thus all the noisy samples is classified as clean or noisy as the preliminary result of first stage.

Entropy-based Aggregation of Rare Classes However, the performance of the proposed feature-based clustering can be unstable when the sizes of some classes are small and not sufficient for binary classification, which often happens in real-world datasets that have large numbers of classes. Addressing this issue, we propose to aggregate rare classes that struggle with the proposed binary classification. Specifically, we set a class aggregate threshold θ_{agg} and calculate the average prediction entropy of the samples for each class c as:

$$\mathrm{Ent}(c) = -\frac{1}{N_c} \sum_{i=1}^{N_c} \sum_{j=1}^{B} p_i^j \log p_i^j, \tag{3}$$

where N_c is the number of samples for class c, $B = 2$ indicates the binary classification of clean versus noisy samples, p_i^j represents the output probability that a sample x_i belongs to class j, *i.e.*, clean and noisy probability. Samples of class c that satisfy $\mathrm{Ent}_c > \theta_{agg}$ are aggregated and treated as a single class to facilitate our feature-based clustering.

3.3 Consistency-based Classification

As Fig 1 shows, challenging clean samples are usually near the ground truth class boundaries in the feature space, which can be identified by the consistency between two independently trained classifiers G_1 and G_2 that have different

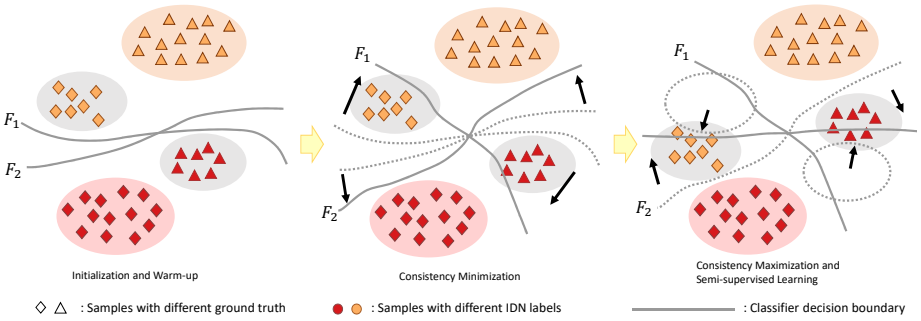


Fig. 4: The procedure of the consistency-based classification. At the beginning, two classifiers has different prediction due to different initialization. Then the prediction consistency between two classifiers is minimized to identify the ambiguous noisy samples near the decision boundary. At the third steps, feature extractor is trained to maximize the consistency and the semi-supervised loss further revises both feature extractor and classifiers.

decision boundaries. Therefore, by replacing the classifier G with G_1 and G_2 in our network, we can get two corresponding predictions p_x^1 and p_x^2 of the same sample x. Then, we define and calculate the consistency between G_1 and G_2 on x as:

$$D(p^1, p^2) = \sum_{i=1}^{C} |p_i^1 - p_i^2|, \tag{4}$$

where x is omitted for simplicity and C is the number of classes, *i.e.* the dimension of p_x^1 and p_x^2. We measure the discrepancy with $L1$ norm following [30].

Consistency Minimization Regularization Although being independently trained, G_1 and G_2 share the same training dataset and the same loss function, leading to a non-negligible risk that the corresponding two predictions are identical or very similar. To minimize such a risk, we propose to incorporate a regularization loss on G_1 and G_2 that aims to minimize their consistency:

$$L_{\text{min}} = -\lambda_{\text{min}} \sum_{i=1}^{N} D^*(p_{x_i}^1, p_{x_i}^2), \tag{5}$$

where N is the number of samples and λ_{min} controls the strength,

$$D^*(p_x^1, p_x^2) = w_{C_x} \sum_{i=1}^{C} |p_i^1 - p_i^2|, \tag{6}$$

where x is omitted on the right side for simplicity and w_{C_x} is the frequency of samples x's noisy category C_x. w_{C_x} is used to counter the class imbalance problem that often happens in real-world datasets. As the GMM model in the first stage does not guarantee the inter-class balance in the clean set, w_{C_x} explicitly

increases the weight of classes with more samples in consistency minimization and thus more samples are filtered out.

Consistency Maximization Regularization Solely using the minimization regularization might impair the model performance because the consistency of samples with correct labels are also minimized, and ideally two classifiers should output the same prediction for each sample. Therefore, we propose to add a consistency maximization loss on the feature extractor F to constrain the network:

$$L_{\max} = \lambda_{\max} \sum_{i=1}^{N} D^*(p^1_{x_i}, p^2_{x_i}), \tag{7}$$

where $\lambda_{\max}$ controls the strength. Furthermore, the maximization of $L_{\max}$ forces the feature extractor to separate the ambiguous features and thus complements semi-supervised training. As shown in the third step of Fig. 4, the feature extractor maximizes the consistency by pushing the samples with small consistency towards clean labeled data, and semi-supervised learning tries to gather the the feature of similar samples.

3.4 Training Procedure

Based on the discussions in Sec. 3.2 and Sec. 3.3, we propose to train our model by repeating the following four steps for each epoch.

Initialization Before training, we following [17] and warm up our model including the two classifiers for several epochs with all noisy labels, where steps 1 and 2 belong to our feature-based clustering (Stage 1), and steps 3 and 4 belong to our consistency-based classification (Stage 2).

Step-1 We first extract the features of noisy data and calculate the class-wise feature centers according to Eq. 1. Then, we calculate the cosine similarities between features and the center of noisy labels of each sample using Eq. 2.

Step-2 We perform a binary (noisy vs. clean) classification to samples by applying class-wise Gaussian Mixture Model (GMM) according to the cosine similarities obtained in Step-1. We label the GMM component with a larger mean as "clean". Then, we select the samples with clean probabilities higher than a threshold θ as our primary clean set S^1_{clean} and the rest samples as the noisy set S^1_{noisy}.

Step-3 We first fix the feature extractor and train the two classifiers to minimize their consistency according to Eq. 5 for $N_{\max}$ iterations using S^1_{clean}. Then, we evaluate the consistency of all samples in S^1_{clean}. Similar to Step-2, we apply a GMM model to the consistencies and select the samples with small mean as clean set S^2_{clean}. The rest samples are merged with S^1_{noisy} as S^2_{noisy}.

Step-4 With S^2_{clean} and S^2_{noisy} obtained as above, we optimize our model with a supervised loss on S^2_{clean} and a semi-supervised loss on S^2_{noisy}:

$$L = L_{\mathcal{X}} + \lambda_{\mathcal{U}} L_{\mathcal{U}} \tag{8}$$

where S^2_{clean} and S^2_{noisy} are used as labeled set $\mathcal{X}$ and unlabeled set $\mathcal{U}$ respectively, and $\lambda_{\mathcal{U}}$ balances the trade-off between $L_{\mathcal{X}}$ and $L_{\mathcal{U}}$. In addition, we add

Table 1: Comparison of test accuracies (%) using different methods on CIFAR10 and CIFAR100 with part-dependent label noise. Results of other methods are copied from CAL[46]. Our method outperforms all previous methods in all settings.

Method	*Inst. CIFAR10*			*Inst. CIFAR100*		
	$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.6$	$\eta = 0.2$	$\eta = 0.4$	$\eta = 0.6$
CE (Standard)	85.45±0.57	76.23±1.54	59.75±1.30	57.79±1.25	41.15±0.83	25.68±1.55
Forward T [27]	87.22±1.60	79.37±2.72	66.56±4.90	58.19±1.37	42.80±1.01	27.91±3.35
L_{DMI} [39]	88.57±0.60	82.82±1.49	69.94±1.31	57.90±1.21	42.70±0.92	26.96±2.08
L_q [42]	85.81±0.83	74.66±1.12	60.76±3.08	57.03±0.27	39.81±1.18	24.87±2.46
Co-teaching [7]	88.87±0.24	73.00±1.24	62.51±1.98	43.30±0.39	23.21±0.57	12.58±0.51
Co-teaching+ [41]	89.80±0.28	73.78±1.39	59.22±6.34	41.71±0.78	24.45±0.71	12.58±0.51
JoCoR [34]	88.78±0.15	71.64±3.09	63.46±1.58	43.66±1.32	23.95±0.44	13.16±0.91
Reweight-R [37]	90.04±0.46	84.11±2.47	72.18±2.47	58.00±0.36	43.83±8.42	36.07±9.73
Peer Loss [20]	89.12±0.76	83.26±0.42	74.53±1.22	61.16±0.64	47.23±1.23	31.71±2.06
CORES^2 [6]	91.14±0.46	83.67±1.29	77.68±2.24	66.47±0.45	58.99±1.49	38.55±3.25
DivideMix[17]	93.33±0.14	**95.07±0.11**	85.50±0.71	79.04±0.21	76.08±0.35	46.72±1.32
CAL[46]	92.01±0.75	84.96±1.25	79.82±2.56	69.11±0.46	63.17±1.40	43.58±3.30
Ours	**93.68±0.12**	94.97±0.09	**94.95±0.11**	**79.61±0.19**	**76.58±0.25**	**59.40±0.46**

additional consistency maximization regularization (Eq. 7) to the feature extractor during training.

4 Experiment

In this section, we will validate the effectiveness of our method on several benchmark datasets with different kinds of IDNs (*i.e.* synthetic and real-world ones) and different numbers of classes.

4.1 Datasets

Synthetic IDN Datasets. Following previous studies on learning with IDN [46], our synthetic IDN datasets are created by adding two kinds of synthetic noise to CIFAR-10 and CIFAR-100 datasets [16], where CIFAR-10 contains 50,000 training images and 10,000 testing images from 10 different classes, CIFAR-100 contains 50,000 training images and 10,000 testing images from 100 classes. Specifically, we use two kinds of synthetic IDN in our experiment:

- Part-dependent label noise [36], which draws insights from human cognition that humans perceive instances by decomposing them into parts and estimates the IDN transition matrix of an instance as a combination of the transition matrices of different parts of the instance.
- Classification-based label noise [3], which adds noise by i) collecting the predictions of each sample in every epoch during the training of a CNN classifier; ii) averaging the predictions and locate the class label with largest prediction probability other than the ground truth one for each instance as its noisy label; iii) flipping the labels of the samples whose largest probabilities falls in the top $r\%$ of all samples, where r is a user-defined hyper-parameter.

Table 2: Classification accuracies on the (clean) test set of Clothing1M. Results of other method are copied from CAL[46]. Our method achieves state-of-the-art performance.

Method	Accuracy
CE (standard)	68.94
Forward T [27]	70.83
Co-teaching [7]	69.21
JoCoR [34]	70.30
L_{DMI} [39]	72.46
PTD-R-V[36]	71.67
DivideMix[17]	74.76
CORES^2 [6]	73.24
CAL[46]	74.17
Ours	**75.40**

Real-world IDN Datasets. Following [17], we use Clothing1M [38] and Webvision 1.0 [18] to evaluate our method:

- Clothing1M is a large scale dataset containing more than 1 million images of 14 kinds of clothes. As aforementioned, Clothing1M is highly imbalanced with its noise validated as IDN according to [3]. In our experiments, we use its noisy training set which contains 1 million images and report the performance on test set.
- Webvision is a large scale dataset which contains 2.4 million images from 1000 classes that are crawled from the web as ImageNet ILSVRC12 did. Following previous works [2,17], we compare baseline methods on the first 50 classes of the Google image subset, and report the top-1 and top-5 performance on both Webvision validation set and ImageNet ILSVRC12.

4.2 Implementation Details

We follow DivideMix [17] and use MixMatch [1] for semi-supervised learning. For experiments on CIFAR-10 and CIFAR-100, we use ResNet-34 [11] as the feature extractor following [46]. We use similar hyperparameters to [17] across all 3 settings of CIFAR-10 and CIFAR-100 respectively. We train our model using a SGD optimizer with a momentum of 0.9 and a weight decay parameter of 0.0005. The learning rate is set as 0.02 in the first 150 epochs and reduced to 0.002 in the following 150 epochs. The warm up period is set as 10 epochs for CIFAR-10 and 15 epochs for CIFAR-100 respectively. For Clothing1M, we follow previous studies and use ImageNet pretrained ResNet-50 as the backbone. We train the model for 80 epochs. We set the learning rate as 0.002 in the beginning and reduce it to 0.0002 after 40 epochs of training. For Webvision 1.0, we follow [17] and use the Inception-Resnet v2[32] as the backbone. We train the model for 120 epochs. We set the learning rate as 0.01 in the first 50 epoch and 0.001 for the rest of the training.

Table 3: Classification accuracies (%) on CIFAR-10 with classification-based label noise of different noise ratios. Our method outperforms all previous ones in all settings.

Method	10%	20%	30%	40%
CE	91.25 ±0.27	86.34 ±0.11	80.87 ±0.05	75.68 ±0.29
Forward[27]	91.06 ±0.02	86.35 ±0.11	78.87 ±2.66	71.12 ±0.47
Co-teaching[7]	91.22 ±0.25	87.28 ±0.20	84.33 ±0.17	78.72 ±0.47
GCE[42]	90.97 ±0.21	86.44 ±0.23	81.54 ±0.15	76.71 ±0.39
DAC[33]	90.94 ±0.09	86.16 ±0.13	80.88 ±0.46	74.80 ±0.32
DMI[39]	91.26 ±0.06	86.57 ±0.16	81.98 ±0.57	77.81 ±0.85
SEAL[3]	91.32 ±0.14	87.79 ±0.09	85.30 ±0.01	82.98 ±0.05
Ours	**91.39** ±0.08	**88.36** ±0.11	**86.92** ±0.68	**84.18** ±0.40

4.3 Experimental Results

CIFAR-10 and CIFAR-100 As aforementioned, we evaluate our method on two kinds of IDN as follows:

- *Part-dependent label noise.* To facilitate a fair comparison, we borrow the noise used in CAL [46] and follow CAL to test the performance of our method against 6 different settings, whose noise ratios vary between 0.2 and 0.6. As Table 1 shows, our method outperforms previous methods in five in six settings, especially when the noise ratio and class number increase. For example, the improvement of CIFAR-100 with $\eta = 0.6$ is over 10%.
- *Classification-based label noise.* Following [3], we test our method against four different noise ratios, 10%, 20%, 30% and 40%. To facilitate a fair comparison, we borrow the same noise from SEAL [3]. Note that compared to the aforementioned part-dependent label noise, the classification-based label noise used in this experiment is more challenging as it is generated by a CNN-based model. As Table 3 shows, our method still outperforms previous methods in all four different settings. Similar as above, the improvement of our method becomes higher as the noise ratio increases, which demonstrates the effectiveness of our method under different kinds of IDNs.

Clothing1M As aforementioned, Clothing1M contains over 1 million images from 14 classes collected from Internet, which makes it ideal to evaluate how different LNL methods perform against large-scale image datasets. As Table 2 shows, our method outperforms all previous methods and achieves the state-of-the-art performance. Compared to DivideMix [17], we further improve the accuracy by 0.64%.

Table 4: Classification accuracies (%) on (mini) Webvision and ILSVRC12. Numbers denote top-1 (top-5) accuracy (%) on the WebVision and the ImageNet ILSVRC12 validation sets.

Method	WebVision		ILSVRC12	
	top1	top5	top1	top5
F-correction [28]	61.12	82.68	57.36	82.36
Decoupling [24]	62.54	84.74	58.26	82.26
D2L [23]	62.68	84.00	57.80	81.36
MentorNet [13]	63.00	81.40	57.80	79.92
Co-teaching [8]	63.58	85.20	61.48	84.70
Iterative-CV [2]	65.24	85.34	61.60	84.98
DivideMix[17]	77.32	91.64	75.20	90.84
NGC[17]	79.16	91.84	74.44	91.04
Ours	**79.36**	**93.64**	**76.08**	**93.86**

Table 5: Ablation study on our Feature Clustering (Stage 1) and Consistency Classification (Stage 2). The models with neither stages are trained with cross-entropy loss (*i.e.* CE baseline).

Dataset	Feature Clustering	Consistency Classification	Accuracy
CIFAR-100 (μ=0.6)			25.68
	✓		53.60
		✓	51.41
	✓	✓	59.40
Clothing1M			68.94
	✓		73.32
		✓	74.26
	✓	✓	75.40

Webvision and ImageNet ILSVRC12 As Table 4 shows, our method achieves better performance on both top-1 and top-5 accuracy on ILSVRC12 and Webvision. The higher improvement on ILSVRC12 suggests that our method is more robust to the domain difference and can generalize better.

4.4 Ablation Study

We conduct an ablation study on the two stages of our method. Specifically, we provide the performance of our method on both CIFAR-100, a synthetic IDN dataset with noise ratio $\eta = 0.6$ and Clothing1M, a highly-imbalanced dataset with real-world IDN. We also compare our method to standard CE baseline (*i.e.* neither stages are applied). As Table 5 shows, our method benefits from each stage in terms of the performance on both datasets, and achieves the best results when both stages are employed.

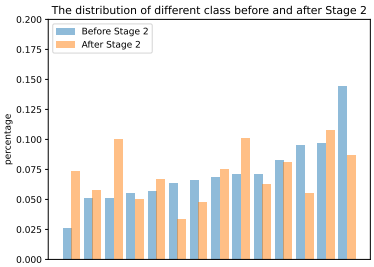


Fig. 5: The distributions of different classes in the validation set of Clothing1M before and after the consistency-based classification (Stage 2). After our consistency-based classification, the distribution becomes more balanced.

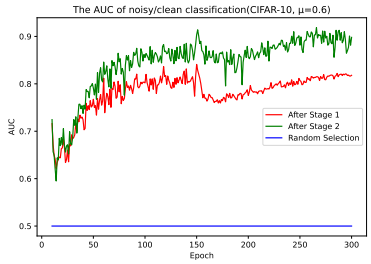


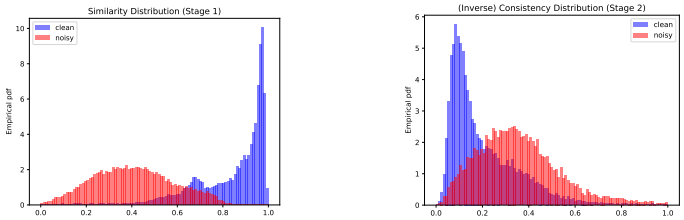
Fig. 6: The AUC of noisy vs. clean classification of our method. The second stage steadily improve the AUC of classification. The performance drop at 150 epoch is due to a learning rate change.

4.5 Performance against Class Imbalance

We select the highly-imbalanced Clothing1M to test the performance of our method against class imbalance. Specifically, we are concerned on the distribution (proportion of class-wise sample number w.r.t the whole dataset) changes of all 14 classes within our selected clean samples before and after our consistency-based classification. Since Clothing1M does not contain the ground truth labels for its noisy training set, we mix some samples from its validation set that contains both clean and noisy labels with the original noisy training set, and report the distributions of the validation samples. As Fig. 5 shows, the percentages of most of the rare classes increase after our consistency-based classification, while the percentages of the rich classes decrease. In addition, we observed biggest changes occur in the rarest and richest classes.

4.6 AUC of Noisy vs. Clean Classification

Given the prediction probabilities of stage 1 and stage 2, we calculate the area under curve (AUC) of our noisy vs. clean classification on CIFAR-10 with a noise ratio of 0.6. As Fig. 6 shows, compared to the performance of random selection,



(a) (b)

Fig. 7: The probability distribution function of clean/noisy samples respectively for CIFAR-10 (μ=0.6). The range of statistics is normalized to 0 to 1. (a) The similarity distribution of stage 1. (b) The (inverse) consistency distribution of stage 2.

both stages of our method can improve the AUC of classification, and the second stage further improve the AUC over the first stage. In addition, it can be observed that the accuracy of noisy vs. clean is improved as the training progresses. The performance decrease occurred around 150 epoch is due to a 0.1-fold decrease of the learning rate. Beside, we provide the probability distribution function of similarity and consistency in Fig. 7. Both metrics are effective in distinguishing clean and noisy samples.

5 Conclusion

In this paper, we propose a two-stage method to address the problem of learning with instance-dependent noisy labels in the presence of inter-class imbalance problem. In the first stage, we identify "easy" clean samples that are close to the class-wise prediction centers using a class-level feature clustering procedure. We also address the class imbalance problem by augmenting the clustering with an entropy-based rare class aggregation technique. In the second stage, we further identify the remaining "difficult" clean samples that are close to the ground truth class boundary based on the consistency of two classifier heads. We conducted extensive experiments on several challenging benchmarks to demonstrate the effectiveness of the proposed method.

Acknowledgements

This work was supported in part by the Guangdong Basic and Applied Basic Research Foundation (No.2020B1515020048), in part by the National Natural Science Foundation of China (No.61976250, No.U1811463), in part by the Hong Kong Research Grants Council through Research Impact Fund (Grant R-5001-18), and in part by the Guangzhou Science and technology project (No.202102020633).

References

1. Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C.: Mixmatch: A holistic approach to semi-supervised learning. arXiv preprint arXiv:1905.02249 (2019)
2. Chen, P., Liao, B.B., Chen, G., Zhang, S.: Understanding and utilizing deep neural networks trained with noisy labels. In: International Conference on Machine Learning. pp. 1062–1070. PMLR (2019)
3. Chen, P., Ye, J., Chen, G., Zhao, J., Heng, P.A.: Beyond class-conditional assumption: A primary attempt to combat instance-dependent label noise. arXiv preprint arXiv:2012.05458 (2020)
4. Chen, P., Ye, J., Chen, G., Zhao, J., Heng, P.A.: Robustness of accuracy metric and its inspirations in learning with noisy labels. arXiv preprint arXiv:2012.04193 (2020)
5. Cheng, H., Zhu, Z., Li, X., Gong, Y., Sun, X., Liu, Y.: Learning with instance-dependent label noise: A sample sieve approach. arXiv preprint arXiv:2010.02347 (2020)
6. Cheng, H., Zhu, Z., Li, X., Gong, Y., Sun, X., Liu, Y.: Learning with instance-dependent label noise: A sample sieve approach. In: International Conference on Learning Representations (2021)
7. Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., Sugiyama, M.: Co-teaching: Robust training of deep neural networks with extremely noisy labels. In: Advances in Neural Information Processing Systems. pp. 8527–8537 (2018)
8. Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I.W., Sugiyama, M.: Co-teaching: Robust training of deep neural networks with extremely noisy labels. In: Advances in Neural Information Processing Systems. pp. 8536–8546 (2018)
9. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)
10. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. pp. 1026–1034 (2015)
11. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778 (2016)
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 (2018)
13. 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. In: International Conference on Machine Learning. pp. 2304–2313. PMLR (2018)
14. Kim, Y., Yun, J., Shon, H., Kim, J.: Joint negative and positive learning for noisy labels. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9442–9451 (2021)
15. Konstantinov, N., Lampert, C.: Robust learning from untrusted sources. In: International Conference on Machine Learning. pp. 3488–3498. PMLR (2019)
16. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images. Tech. rep., Citeseer (2009)
17. Li, J., Socher, R., Hoi, S.C.: Dividemix: Learning with noisy labels as semi-supervised learning. In: International Conference on Learning Representations (2020), `https://openreview.net/forum?id=HJgExaVtwr`

18. 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 (2017)
19. Liu, T., Tao, D.: Classification with noisy labels by importance reweighting. IEEE Transactions on Pattern Analysis and Machine Intelligence **38**(3), 447–461 (2015)
20. Liu, Y., Guo, H.: Peer loss functions: Learning from noisy labels without knowing noise rates. In: Proceedings of the 37th International Conference on Machine Learning. ICML '20 (2020)
21. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3431–3440 (2015)
22. Lu, Y., Bo, Y., He, W.: Co-matching: Combating noisy labels by augmentation anchoring. arXiv preprint arXiv:2103.12814 (2021)
23. Ma, X., Wang, Y., Houle, M.E., Zhou, S., Erfani, S., Xia, S., Wijewickrema, S., Bailey, J.: Dimensionality-driven learning with noisy labels. In: International Conference on Machine Learning. pp. 3355–3364. PMLR (2018)
24. Malach, E., Shalev-Shwartz, S.: Decoupling "when to update" from "how to update". In: Advances in Neural Information Processing Systems. pp. 960–970 (2017)
25. Nguyen, D.T., Mummadi, C.K., Ngo, T.P.N., Nguyen, T.H.P., Beggel, L., Brox, T.: Self: Learning to filter noisy labels with self-ensembling. arXiv preprint arXiv:1910.01842 (2019)
26. Nishi, K., Ding, Y., Rich, A., Hollerer, T.: Augmentation strategies for learning with noisy labels. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8022–8031 (2021)
27. Patrini, G., Rozza, A., Krishna Menon, A., Nock, R., Qu, L.: Making deep neural networks robust to label noise: A loss correction approach. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1944–1952 (2017)
28. Patrini, G., Rozza, A., Krishna Menon, A., Nock, R., Qu, L.: Making deep neural networks robust to label noise: A loss correction approach. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1944–1952 (2017)
29. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28** (2015)
30. Saito, K., Watanabe, K., Ushiku, Y., Harada, T.: Maximum classifier discrepancy for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3723–3732 (2018)
31. Shen, Y., Sanghavi, S.: Learning with bad training data via iterative trimmed loss minimization. In: International Conference on Machine Learning. pp. 5739–5748. PMLR (2019)
32. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: Thirty-first AAAI conference on artificial intelligence (2017)
33. Thulasidasan, S., Bhattacharya, T., Bilmes, J., Chennupati, G., Mohd-Yusof, J.: Combating label noise in deep learning using abstention. arXiv preprint arXiv:1905.10964 (2019)
34. Wei, H., Feng, L., Chen, X., An, B.: Combating noisy labels by agreement: A joint training method with co-regularization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13726–13735 (2020)

35. Xia, X., Liu, T., Han, B., Wang, N., Deng, J., Li, J., Mao, Y.: Extended t: Learning with mixed closed-set and open-set noisy labels. arXiv preprint arXiv:2012.00932 (2020)
36. Xia, X., Liu, T., Han, B., Wang, N., Gong, M., Liu, H., Niu, G., Tao, D., Sugiyama, M.: Part-dependent label noise: Towards instance-dependent label noise. In: Advances in Neural Information Processing Systems. vol. 33, pp. 7597–7610 (2020)
37. Xia, X., Liu, T., Wang, N., Han, B., Gong, C., Niu, G., Sugiyama, M.: Are anchor points really indispensable in label-noise learning? In: Advances in Neural Information Processing Systems. pp. 6838–6849 (2019)
38. 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)
39. Xu, Y., Cao, P., Kong, Y., Wang, Y.: L_dmi: A novel information-theoretic loss function for training deep nets robust to label noise. In: Advances in Neural Information Processing Systems. pp. 6222–6233 (2019)
40. Yao, Y., Sun, Z., Zhang, C., Shen, F., Wu, Q., Zhang, J., Tang, Z.: Jo-src: A contrastive approach for combating noisy labels. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5192–5201 (2021)
41. Yu, X., Han, B., Yao, J., Niu, G., Tsang, I.W., Sugiyama, M.: How does disagreement help generalization against label corruption? arXiv preprint arXiv:1901.04215 (2019)
42. 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)
43. Zheltonozhskii, E., Baskin, C., Mendelson, A., Bronstein, A.M., Litany, O.: Contrast to divide: Self-supervised pre-training for learning with noisy labels. arXiv preprint arXiv:2103.13646 (2021)
44. Zheng, G., Awadallah, A.H., Dumais, S.: Meta label correction for noisy label learning. In: Proceedings of the 35th AAAI Conference on Artificial Intelligence (2021)
45. Zhou, H.Y., Chen, X., Zhang, Y., Luo, R., Wang, L., Yu, Y.: Generalized radiograph representation learning via cross-supervision between images and free-text radiology reports. Nature Machine Intelligence **4**, 32–40 (2022)
46. Zhu, Z., Liu, T., Liu, Y.: A second-order approach to learning with instance-dependent label noise. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10113–10123 (2021)
47. Zhu, Z., Song, Y., Liu, Y.: Clusterability as an alternative to anchor points when learning with noisy labels. arXiv preprint arXiv:2102.05291 (2021)