Learning with Structural Labels for Learning with Noisy Labels

Noo-ri Kim* Jin-Seop Lee* Jee-Hyong Lee†
Sungkyunkwan University, Suwon, South Korea
{pd99j, wlstjq0602, john}@skku.edu

Abstract

Deep Neural Networks (DNNs) have demonstrated remarkable performance across diverse domains and tasks with large-scale datasets. To reduce labeling costs for large-scale datasets, semi-automated and crowdsourcing labeling methods are developed, but their labels are inevitably noisy. Learning with Noisy Labels (LNL) approaches aim to train DNNs despite the presence of noisy labels. These approaches utilize the memorization effect to select correct labels and refine noisy ones, which are then used for subsequent training. However, these methods encounter a significant decrease in the model's generalization performance due to the inevitably existing noise labels. To overcome this limitation, we propose a new approach to enhance learning with noisy labels by incorporating additional distribution information—structural labels. In order to leverage additional distribution information for generalization, we employ a reverse k-NN, which helps the model in achieving a better feature manifold and mitigating overfitting to noisy labels. The proposed method shows outperformed performance in multiple benchmark datasets with IDN and real-world noisy datasets.

1. Introduction

Deep neural networks (DNNs) have achieved high performance in various domains such as computer vision [13, 14, 20, 25, 29, 52], natural language processing [3, 7, 12, 40], and signal processing [18, 48, 50] across diverse tasks. To achieve the high performance of DNNs, well-curated large-scale datasets are necessary. However, labeling large-scale data requires a significant cost. In order to reduce these labeling costs, semi-automated [35, 42] and crowdsourcing methods [28] are used, but their labels are inevitably noisy. When trained on data with noisy labels, DNNs suffer degraded performance because they tend to easily overfit noisy labeled samples [72]. To address these issues, methods for learning with noisy labels (LNL), which aim to effectively train DNN models in environments with noisy labeled samples, have been proposed [1, 15, 17, 23, 31, 41, 45, 75].

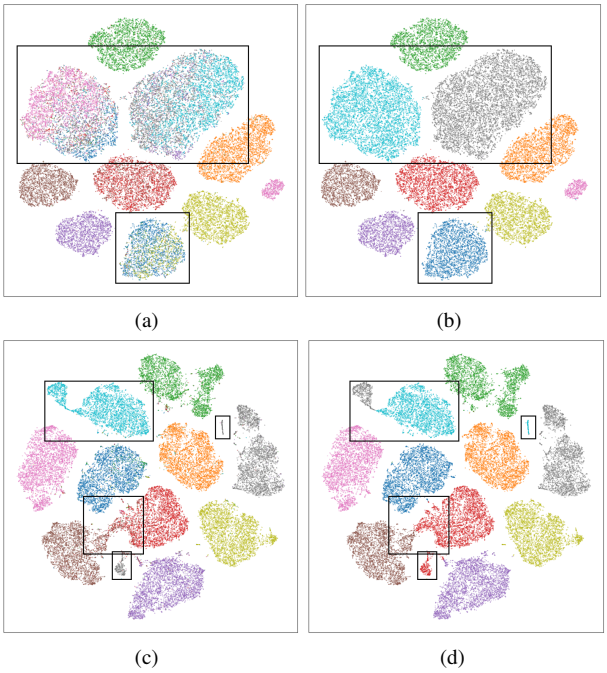


Figure 1. Comparison of t-SNE visualization on CIFAR10 IDN at noise rate 0.50. (a) DivdeMix on training samples with golden labels. (b) DivdieMix on training samples with predicted labels. (c) SSR on training samples with golden labels. (d) SSR on training samples with predicted labels. The black-outlined box represents areas where samples overlap, similar samples are learned as different classes, or clusters are not separable.

Most methods for learning with noisy labels (LNL) are based on the fact that DNNs tend to learn from clean samples with simple patterns earlier than from wrong-labeled samples [1, 69]. Based on the observation, most approaches adopted an iterative process. Initially, they trained the model with the entire dataset. The model would learned some patterns from clean samples rather than noisy samples. Then, they selected confident samples based on the

*Equal contribution
†Corresponding author

output of the trained model, most of which would be clean samples. Subsequently, they retrained the model with the selected samples. They repeat the selection and training. Consequently, the model can avoid noisy samples and focus on clean samples. Previous approaches proposed various clean sample selection methods based on the model prediction, such as the small-loss trick [23] and the k-Nearest Neighbor (k-NN) method [9, 16]. Additionally, relabeling methods are utilized to leverage a larger number of clean samples by modifying model outputs deemed unreliable. These learning strategies performed quite well in a noisy environment, but they still have a critical limitation—they heavily depend on the model's predictions for relabeling and selection.

Essentially, the models guide themselves based on their own predictions in a noisy environment. If an incorrectly predicted sample is chosen, the model may learn from it in the subsequent training step. Therefore, in this self-guided process, inevitable inaccurate predictions arise from learning based on previous inaccuracies. By learning from these inevitable noisy labels, the model not only degrades its generalization performance but also declines in its ability to extract features, making the feature manifold more complex.

Figure 1 shows that the learned feature manifolds are complex due to a degradation of generalization performance. It presents t-SNE [60] visualization for DivideMix [31], SSR [15]. Figures 1a and 1b display features of training samples extracted by DivideMix with golden labels and predictions, respectively. In DivideMix, a considerable number of samples from different classes exist in a single cluster, and the model tends to predict them with the same label, incorrectly. In Fig. 1c and 1d, SSR shows better results compared to DivideMix, but clusters have more complex forms. Furthermore, SSR incorrectly predicts some samples, and it seems that classes are not separable. They made efforts to improve generalization by incorporating additional objectives, such as consistency regularization [15] and unsupervised loss [17, 31, 75], but still demonstrate poor generalization performance.

It is a fact that deep learning models trained with all cleanly labeled samples exhibit good generalization behavior [44], and their feature distributions are well-discriminated by class. These well-generalized models have preserved the following structural characteristics: 1) When two samples are similar, it is highly probable that they share the same label. 2) If certain samples form a cluster, they are likely to have the same label. These are also consistent with the smoothness/cluster assumptions [4, 61, 76] in the context of semi-supervised learning. We aim to develop a method for training a LNL model by consider the characteristics of well-generalized models. In order to incorporate the characteristics, we aim to extract structural information based on the distribution of the data and directly integrate it into the training process.

In this paper, we present a simple approach to enhance learning with noisy labels by incorporating additional distribution information, structural labels. If a model preserves the structural assumption, it can avoid overfitting to noisy labels and achieve better generalized. By extracting the structural information of data and integrating it into model training, we can efficiently train models with minimized bias towards noisy samples. To extract structural labels, we estimate the data distribution based on Reverse k-Nearest Neighbor (Reverse k-NN) [57]. And then, the structural labels are learned with cross-entropy based on strong augmentation [10] and mixup [73] methods, which help to avoid overfitting noisy samples.

To verify our proposed method, we conduct experiments on IDN and real-world noisy datasets including CIFAR10 and CIFAR100 with IDN [66], CIFAR-N [63], Animal-10N [53], Red Mini-Imagenet from CNWL [24], Mini-WebVision [35], and Imagenet ILSVRC12 [11]. Our method shows the state-of-the-art performance on various datasets.

2. Related Work

Deep neural networks (DNNs) tend to overfit to label datasets so that they also overfit to noisy label samples and it leads to degraded performance. Many studies have been conducted to learn datasets with noisy labels. To prevent overfitting to noisy labels and learn effectively from datasets with noisy labels, early studies utilized a given small set of clean samples and relabeled noisy samples based on their modeling [5, 30, 36, 59, 62, 67]. Subsequently, instead of using a given small set of clean samples, several studies focus to distinguish clean samples from the given noisy datasets [1, 15, 17, 26, 31, 51, 58, 75].

These methods rely on DNNs initially learn from cleanly labeled samples with simple patterns before overfitting to noisy label samples [72], and use the small-loss trick to select clean-label samples [17, 23, 31, 51, 75]. DivideMix [31] treats samples with small losses as labeled samples and the others as unlabeled then trains the model using MixMatch [2], a semi-supervised learning technique. C2D [75] leverages self-supervised models to tackle the instability during the warm-up phase of DivideMix, and InstanceGM [17] proposes a graphical modeling approach to address the more realistic noise, Instance-Dependent Noise (IDN).

Contrary to the loss-based approaches, some methods utilize the model's feature representation [15, 34, 64, 71]. TopoFilter [64] proposes a label noise filtering strategy based on spatial topological patterns, and SSR [15] leverages the model's high-confident predictions for relabeling samples and employs a k-NN classifier [9] within the feature representation space for selecting clean samples.

In conclusion, recent LNL methods mainly focus on relabeling and selecting based on their own predictions. However, it is challenging to expect good generalization performance due to the excessive reliance on the predictions of a model that may have errors.

3. Proposed Method

Since existing methods heavily depend on uncertain models' predictions, they struggle to form good representations and do not preserve structural information. Well-generalized models should preserve the assumption that closer features are likely to be located on the same manifold and have the same label. Therefore, it is necessary not only to get accurate predictions from the model but also to consider the distribution of features.

In this section, we propose *Learning with Structural Label* (LSL), a new approach for learning structural information in LNLs. To learn structural information, we estimate the data distribution based on the Reverse k-NN algorithm, which is less influenced by outlier samples compared to k-NN [21, 49], and calculate the probability of belonging to a class distribution. Then, we extract structural labels l_{st} from the estimated distribution for all samples and train the model using a strong augmentation [10] and mixup [73] strategy. In contrast to traditional relabeling methods, which depend on uncertain models' predictions learned from selected samples with simple patterns, our approach considers the feature distribution for learning on all samples, not only selected samples.

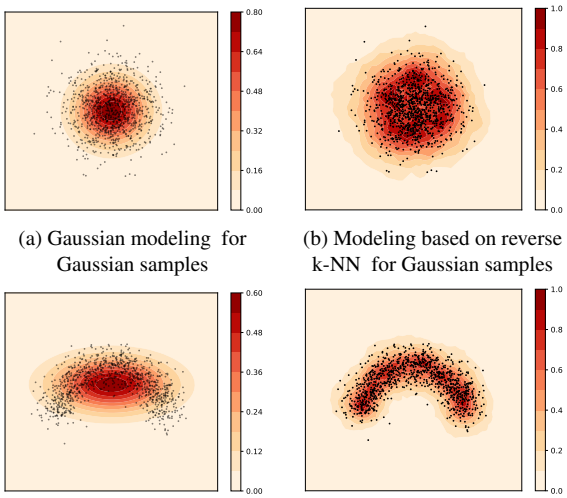
3.1. Structural Labels based on Reverse k-NN

In order to obtain more reliable information from noisy predictions of the model, we define the structural labels, l_{st}, which are the soft labels of each sample based on sample distributions. We first obtain class prediction for each sample from the model, and estimate the distribution of samples predicted as class c. We estimate the probability of x given c, $P(x|c)$, based on the similarity of feature representations and reverse k-NN. We emit k arrows from each sample predicted as class c to its k nearest neighbors including itself. For a sample x, $P(x|c)$ can be approximated as follows:

$$P(x|c) = \frac{\#\text{ of Arrows}_{x,c}}{k \cdot |X^c|} \tag{1}$$

where $\#\text{ of Arrows}_{x,c}$ is the total number of arrows received by x from samples predicted as c, and $|X^c|$ is the total number of samples predicted as c. If x exists in the region where c class samples densely exist, it may receive more arrows, and vice versa. Consequently, when divided by the total number of arrows from c class samples, it is an approximation for $P(x|c)$.

Figure 2 shows estimation examples of 2-d Guassian and Moon data [47]. Figures 2a and 2c show distributions by Gaussian modeling, and 2b and 2d show distributions by our modeling based on reverse k-NN. The Gaussian modeling can efficiently estimates the distributions, but it is hard to accurately model complex data distributions like the Moon dataset. In contrast, the distribution estimation based on reverse k-NN effectively approximates not only the simple distribution, Gaussian, but also the complex data distribution, Moon dataset.



(a) Gaussian modeling for Gaussian samples

(b) Modeling based on reverse k-NN for Gaussian samples

(c) Gaussian modeling for Moon dataset

(d) Modeling based on reverse k-NN for Moon dataset

Figure 2. Comparison of distributions between Gaussian modeling and reverse k-NN modeling in 2-D Gaussian dataset and Moon dataset. Figure 2a shows the Gaussian modeling effectively estimates distribution for Gaussian samples. However, the Gaussian modeling fails to fit the distribution of the Moon dataset in Fig. 2c. Figure 2b and Fig. 2d show Reverse k-NN effectively approximates the data distribution for both Gaussian and Moon datasets.

Based on the feature distribution, we define the structural label of x, which is the soft label representing the probability of c given x:

$$P(c|x) = \frac{P(x|c)P(c)}{\sum_{c \in C} P(x|c)P(c)}. \tag{2}$$

The prior $P(c)$ represents the probability of observing class c, which can be determined from the number of samples in each class:

$$P(c) = \frac{|X^c|}{\sum_{c=1}^{C} |X^c|}. \tag{3}$$

Then, Eq. (2) can be reformulated as follows:

$$P(c|x) = \frac{P(x|c)\,|X^c|}{\sum_{c=1}^{C} P(x|c)\,|X^c|}. \tag{4}$$

From Eqs. (1) and (4), $P(c|x)$ is approximated as follows:

$$P(c|x) = \frac{\#\text{ of Arrows}_{x,c}}{\sum_{c=1}^{C} \#\text{ of Arrows}_{x,c}}. \tag{5}$$

Algorithm 1: Extracting Structural Labels based on reverse k-NN distribution estimation

Input : model encoder θ,
training dataset $\mathcal{X}, \mathcal{Y} = \{(x_i, y_i^r)_{i=1}^N\}$
Parameter: number of reverse nearest neighbors k_{st}
Output : structural labels by *st* $\mathcal{Y}^{st} = \{(y^{st})_{i=1}^N\}$

T = torch.zeros[N, K]
 # K is the number of classes
 # T indicates edge count table
$\mathcal{F}$ = *Normalize*$(\theta(\mathcal{X}))$ *# Normalize all features*
for $i \leftarrow 1$ **to** N **do**
  score = *Cosine*$(f_i, \mathcal{F})$
  indices = score.topk(k_{st})
  # Emit k edges to k_{st} *nearest neighbors*
  T [indices, y_i^r] += 1
  # Propagate its label to k nearest neighbors
end
$\mathcal{Y}^{st}$ = T / T.sum(dim=1, keepdim=True)
 # Aggregate from received labels and normalize these labels
return $\mathcal{Y}^{st}$

Since $P(c|x)$ is estimated from noisy prediction of the model, it can be affected by wrong predictions. However, if we assume that the uniform noise over the sample space, it is clear that the probability is free from the uniform noise. Even though the noise is not uniform, it can be easily smoothed if we use enough large k_{st}. However, if k_{st} is too large, the number of arrows received from samples of classes other than c will increase. It can lead to over-smoothing of l_{st} and lose the structural information. To avoid over-dilution of structural information, we need to choose an appropriate value of k_{st}, and it is discussed in Sec. 5.2.

In conclusion, the structural label of x estimated by reverse k-NN effectively reflects the distribution information of the samples in noisy data environments. Algorithm 1 shows how to obtain the structural labels.

3.2. Learning with Structural Labels (LSL)

To learn structural information, we utilize structural labels l_{st} to effectively reflect feature distribution with SSR [15]. It is one of the LNL methods that considers feature representation. The procedure for the proposed method is described in Algorithm 2.

In Algorithm 2, line 2 indicates relabeling based on model prediction with Eq. (6), and line 3 indicates sample selection based on k-NN.

$$y_i^r = \begin{cases} \text{argmax}_l f(\alpha(x_i)), & \text{if } \max_l f(\alpha(x_i)) > \tau_r \\ y_i, & \text{otherwise} \end{cases} \quad (6)$$

Algorithm 2: **L**earning with **S**tructural **L**abels (LSL)

Input : model encoder θ,
multi-layer neural network ϕ,
training dataset $(\mathcal{X}, \mathcal{Y}) = \{(x_i, y_i)\}_{i=1}^N$
Parameter: sample relabelling threshold τ_r,
feature consistency loss weight λ_{fc},
structural loss weight λ_{st}

while e < epochs **do**
  $\mathcal{Y}^r \leftarrow$ *Sample Relabeling*$(\theta, \mathcal{X}, \tau_r)$ with Eq. (6)
  $\mathcal{X}_{sel}, \mathcal{Y}_{sel}^r \leftarrow$ *Sample Selection*$(\theta, \mathcal{X})$
  # Relabeling and Selection with SSR
  $\mathcal{Y}^{st} \leftarrow$ *Proposed Sample Relabeling*$(\theta, \mathcal{X}, \mathcal{Y}^r)$
  # Structural labels extracted from Algorithm 1
  From $\mathcal{X}_{sel}$, draw a mini-batch $\{(x_{sel,b}, y_{sel,b}^r); b \in (1, ..., B)\}$
  From $\mathcal{X}$, draw a mini-batch $\{(x_b, y_b^{st}); b \in (1, ..., B)\}$
  for $b \leftarrow 1$ **to** B **do**
    $L_{ce} = \text{Mixup}(\theta(A(x_{sel,b})), y_{sel,b}^r)$
    # A is a strong-augmentation
    $L_{fc} = -\text{Cosine}(\phi(\theta(\alpha(x_b))), \phi(\theta(A(x_b))))$
    # α *is a weak-augmentation*
    # Feature consistency learning
    $L_{st} = \text{Mixup}(\theta(A(x_b)), y_b^{st})$
    # Calculate structural loss with y_{st}
    $L = L_{ce} + \lambda_{fc} L_{fc} + \lambda_{st} L_{st}$
    $\theta = \text{SGD}(L, \theta)$
    # Update model parameters by minimizing L
  end
end

where, α indicates the weak-augmentation, and τ_r indicates the relabeling threshold.

In line 5, structural labels l_{st}, which incorporate distribution information, are extracted as described in Sec. 3.1. These structural labels are learned with cross-entropy based on the strong augmentation [10] and mixup [73], following the same manner as learning selected samples in line 10. Consequently, by learning structural labels, the model preserves the structural assumption, leading to an enhancement in the model's generalization performance.

4. Experiments

To verify the effectiveness of our method, we conduct experiments on multiple noisy benchmark datasets such as CIFAR10 and CIFAR100 with IDN [27, 66], CIFAR-N [63]. Additionally, we evalutate our method on real-world noisy datasets such as Animal-10N [53], Red Mini-Imagenet from CNWL [24], Mini-webvision [35], and Imagenet ILSVRC12 [11].

Method	IDN - CIFAR10					IDN - CIFAR100				
	0.20	0.30	0.40	0.45	0.50	0.20	0.30	0.40	0.45	0.50
CE [70]	75.81	69.15	62.45	51.72	39.42	30.42	24.15	21.45	15.23	14.42
Mixup [73]	73.17	72.02	61.56	56.45	48.95	32.92	29.76	25.92	23.13	21.31
Forward [46]	74.64	69.75	60.21	48.81	46.27	36.38	33.17	26.75	21.93	19.27
Reweight [39]	76.23	70.12	62.58	51.54	45.46	36.73	31.91	28.39	24.12	20.23
Decoupling [43]	78.71	75.17	61.73	58.61	50.43	36.53	30.93	27.85	23.81	19.59
Co-teaching [19]	80.96	78.56	73.41	71.60	45.92	37.96	33.43	28.04	25.60	23.97
MentorNet [23]	81.03	77.22	71.83	66.18	47.89	38.91	34.23	31.89	27.53	24.15
DivideMix [31]	94.80	94.60	94.53	94.08	93.04	77.07	76.33	70.80	57.78	58.61
SSR∗ [15]	96.49	96.52	96.33	95.89	94.06	78.84	78.60	76.95	74.98	72.83
InstanceGM [17]	96.68	96.52	96.36	96.15	**95.90**	79.69	79.21	78.47	77.49	77.19
LSL (Ours)	**97.13**	**96.85**	**96.53**	**96.58**	95.81	**80.94**	**79.90**	**78.60**	**78.08**	**77.95**

Table 1. Test accuracy (%) of different methods on CIFAR10-IDN and CIFAR100-IDN [66] under various IDN noise rates. Most of the experimental results are extracted from state-of-the-art methods [15, 17, 31]. **Bold values** indicate the best performances. The reproduced result is marked with ∗.

We describe these datasets in Sec. 4.1 and provide the implementation details in Sec. 4.2. In Sec. 4.3, we compare our approach with state-of-the-art models in various IDN benchmarks and real-world noisy datasets.

4.1. Datasets

CIFAR10 and CIFAR100 both datasets [27] consist of 50,000 training images and 10,000 testing images, and each image has dimensions of 32×32×3. The CIFAR10 has 10 classes, while CIFAR100 has 100 classes. To conduct experiments in the instance-dependent noise (IDN) environment, we obtained noisy labels of the CIFAR10 and CIFAR100 datasets according to [66]. We set the noise rate for IDN settings at values of 0.20, 0.30, 0.40, 0.45, and 0.50.

The CIFAR10N and CIFAR100N datasets [63] consist of human-annotated noisy labels obtained from Amazon Mechanical Turk. CIFAR10N and CIFAR100N datasets include 5 and 2 noisy label types, respectively. In this paper, we evaluate the settings with the highest noise rates, specifically CIFAR10N Worst (CIFAR10N-W, a noise rate of 40.21%) and CIFAR100N Fine (CIFAR100N-F, a noise rate of 40.20%).

Animal-10N dataset [53] consists of 50,000 training images and 10,000 test images, with each image having a resolution of 64×64×3. The dataset has 10 animal categories, including 5 pairs of animals with similar appearances and the estimated noise rate of the dataset is 8%.

Red Mini-Imagenet from CNWL [24] dataset is annotated by 3-5 labeling professionals using Google Cloud Data Labeling Service. The dataset comprises 600 training samples for each of its 100 classes. While the original image size is 84×84×3, we resize it to 32×32×3 for a fair comparison with other experiments [17, 68]. It supports noise ratios ranging from 0% to 80%, and we validate our model on training data with noise ratios of 20%, 40%, 60%, 80%.

WebVision [35] is a large-scale dataset with 2.4 million images collected from the web across 1k categories, and Mini-WebVision is a dataset composed of the initial 50 categories from the WebVision dataset. Mini-WebVision contains 65,994 training images, and we resized the images to 256×256×3. For validation, we use not only Mini-WebVision but also ImageNet ILSVRC12 dataset [11], which shares the same subset of categories.

4.2. Implementation Details

All experiments are conducted using the PyTorch framework on an NVIDIA RTX 3090Ti GPU. Unless otherwise specified, all hyperparameter values mentioned below are followed from SSR [15].

For CIFAR10 and CIFAR100 with IDN datasets, we use a PreAct-ResNet-18 [20] following DivideMix [31]. We use SGD optimizer for 300 epochs with a momentum of 0.9, a weight decay of 5e-4, an initial learning rate of 0.02, and the cosine annealing function as a learning scheduler. We set a relabeling threshold, τ_r, from 0.55 to 0.8 for CIFAR10 and CIFAR100 with IDN as the noise ratio decreases, and the batch size to 128. For CIFAR-N datasets, we set τ_r to 0.8 for CIFAR10-N and CIFAR100-N, and all other details are consistent with CIFAR10/100 with IDN configurations. For Red Mini-Imagenet from CNWL [24], we set τ_r to 0.8, and all other details are consistent with CIFAR10/100 with

Method	CIFAR10N-W	CIFAR100N-F
CE [70]	77.69	55.50
CAL [77]	85.36	61.73
ELR [37]	91.09	66.72
SOP+ [38]	93.24	67.81
DivideMix [31]	92.56	71.13
SSR∗ [15]	93.50	71.99
LSL (Ours)	**94.57**	**74.46**

Table 2. Test accuracy (%) of CIFAR-N [63]. **Bold values** indicate the best performances and the reproduced result is marked with ∗.

IDN configurations.

For Animal-10N, we use a VGG-19 with batch-normalization [55]. We use SGD optimizer for 150 epochs with a momentum of 0.9, a weight decay of 5e-4, an initial learning rate of 0.02, and the step function as a learning scheduler with a reduction factor of 10 and a step size of 50 epochs. We set τ_r to 0.8, and the batch size of 128.

For Mini-WebVision, we use an InceptionResNetv2 [56] following [31]. We use SGD optimizer for 150 epochs with a momentum of 0.9, a weight decay of 1e-4, an initial learning rate of 0.01, and the step function as a learning scheduler with a reduction factor of 10 and a step size of 50 epochs. We set τ_r to 0.8, and the batch size to 64.

For mixup interpolation, we set both α and β to 4 for beta mixture on IDN datasets, CIFAR-N and Red Mini-ImageNet, and to 0.5 on other datasets. We set the consistency loss weight λ_{fc} and the structural loss weight λ_{st} to 1.0, and the number of reverse nearest neighbors k_{st} to 20.

4.3. Experimental Results

The results for CIFAR10 and CIFAR100 with IDN benchmark datasets are shown in Tab. 1. Our proposed method exhibits the best performance in all cases except for CIFAR10 with an IDN noise ratio of 0.5. Even in CIFAR10 with an IDN noise ratio of 0.5, our method demonstrates comparable performance to the baseline, within a margin of 0.1% points. These results demonstrate that our method is consistently effective across various IDN environments.

Tables 2 to 5 show the results for real-world noisy datasets. The performance on CIFAR10N-W and CIFAR100N-F is presented in Tab. 2. For both datasets, which have a noise rate of approximately 40%, our proposed method shows significant improvements by 1.1% and 3.4%, respectively, compared to the baseline methods. These results confirm the effectiveness of our approach, not only with synthetic IDN noise but also with human-annotated real-world noisy labels.

Method	Noise rate			
	0.2	0.4	0.6	0.8
CE [70]	47.36	42.70	37.30	29.79
Mixup [73]	49.10	46.40	40.58	33.58
MentorMix [24]	51.02	47.14	43.80	33.46
FaMUS [68]	51.42	48.06	45.10	35.50
DivideMix [31]	50.96	46.72	43.14	34.50
SSR∗ [15]	52.18	48.96	42.42	33.20
InstanceGM [17]	**58.38**	**52.24**	**47.96**	**39.62**
LSL (Ours)	54.68	49.80	45.46	36.78

Table 3. Test accuracy (%) of Red Mini-Imagenet (CNWL) [24]. **Bold values** indicate the best performances and the reproduced result is marked with ∗.

Method	Test Accuracy (%)
CE [70]	79.4
Nested-Dropout [6]	81.8
CE+Dropout [6]	81.3
SELFIE [54]	81.3
PLC [74]	83.4
Nested-CE [6]	84.1
SSR [15]	88.5
InstanceGM [17]	84.6
LSL (Ours)	**89.1**

Table 4. Test accuracy (%) of ANIMAL-10N [53]. **Bold values** indicate the best performances.

The experimental results on the Red Mini-ImageNet from CNWL dataset are presented in Tab. 3. Our proposed method demonstrates the second-best performances.

Animal-10N is a challenging dataset consisting of 5 pairs of easily confused animals and has a relatively low noise ratio of approximately 8% compared to other real-world noisy labels. For this dataset, our method outperforms other approaches, as shown in Tab. 4.

Lastly, Tab. 5 shows performance on the large-scale WebVision and ImageNet datasets. Our method achieves the best performance on all cases in Top1 accuracy. Especially, our proposed method shows best performance on the Top1 accuracy of ImageNet, and it shows that our method generalizes very well to other datasets.

Method	WebVision		IISVRC2012	
	Top-1	Top-5	Top-1	Top-5
Co-teaching [19]	63.58	85.20	61.48	84.70
ELR+ [37]	77.78	91.68	70.29	89.76
NGC [65]	79.16	91.84	74.44	91.04
LongReMix [8]	78.92	92.32	-	-
RRL [32]	76.30	91.50	73.30	91.20
Sel-CL+ [33]	79.96	92.64	76.84	**93.04**
TCL [22]	79.10	92.30	75.40	92.40
DivideMix [31]	77.32	91.64	75.20	90.84
SSR [15]	80.92	92.80	75.76	91.76
LSL (Ours)	**81.40**	**93.00**	**77.00**	91.84

Table 5. Test accuracy (%) of WebVision [35] and IISVRC2012 [11]. **Bold values** indicate the best performances.

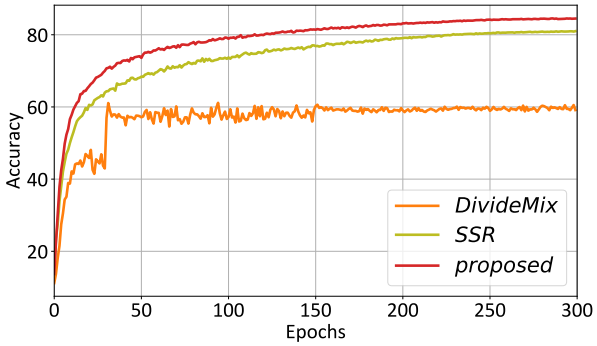
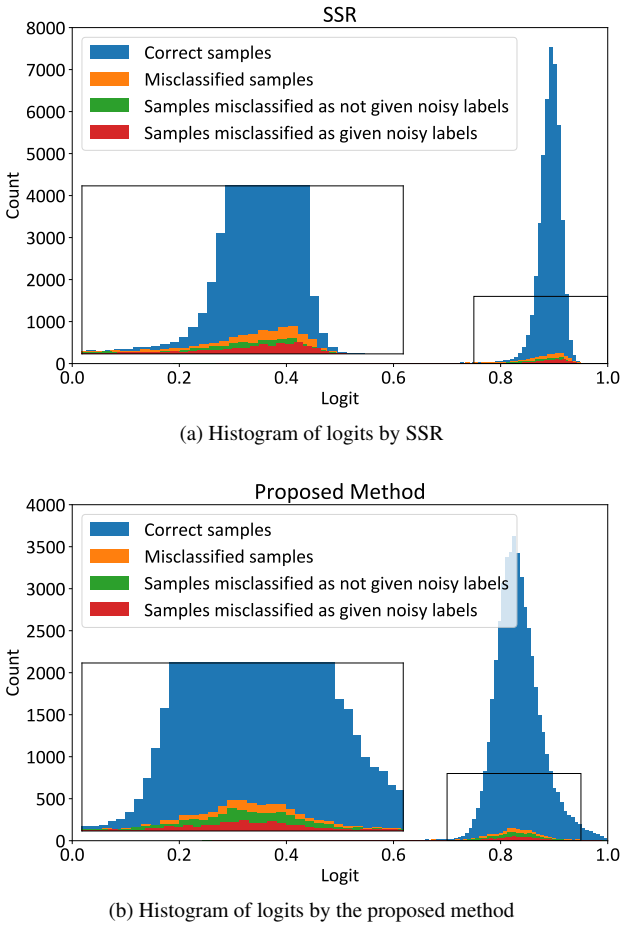


Figure 3. Comparison of training accuracy. Our proposed method outperforms existing approaches. Through the proposed structural labels based on the feature distribution, the model can effectively leverage the entire training dataset.



(a) Histogram of logits by SSR

(b) Histogram of logits by the proposed method

Figure 4. Histogram of Logits for the Training Dataset of CIFAR10 with IDN at noise rate 0.50.

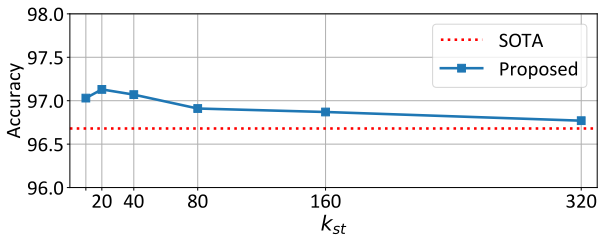
5. Ablation Studies

In this section, we analyze the effectiveness of our proposed method and introduce its details. All ablation studies are conducted on the CIFAR10 with IDN dataset [66].

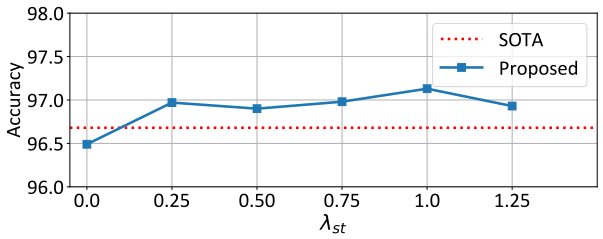
5.1. Effectiveness of learning structural labels

Figure 3 shows the accuracy of training data, the agreement between the model's predictions and the original clean labels as training progresses. The original clean labels are used only for evaluation, not for training. In LNL, the accuracy typically does not reach 100% by the end of training as models cannot fully generalize from noisy training data. Our proposed method shows a higher training accuracy compared to other approaches such as DivideMix [31] and SSR [15]. Existing methods are difficult to accurately relabel and select more samples with complex patterns, so they only learn a fraction of the training data and struggle to generalize. In contrast, our proposed method, which utilizes structural labels l_{st} based on feature distribution, effectively leverages more samples and achieves better generalization performance.

Figure 4 shows the distribution of logits for SSR and our proposed method on CIFAR10 with IDN at noise rate 0.50. In SSR, the total number of misclassified samples is 3238. Among them, 1248 samples are misclassified as given noisy labels, accounting for 0.39 percent of the total misclassified samples. On the other hand, in our method, the total number of misclassified samples is 2107. Among them, the number of misclassified samples as given noisy labels is 646. The percentage of them is 0.31. Thus, our proposed method exhibits fewer total misclassified samples and a lower percentage of misclassified samples as given noisy labels. Also, The logit distribution of misclassified samples with given noise (red) is skewed to the right relative to the logit distribution of misclassified samples (orange) in SSR, whereas



(a) Accuracy with respect to different values of the k_{st}.



(b) Accuracy with respect to different values of the λ_{st}.

Figure 5. Accuracy with respect to hyperparameters on CIFAR10 with IDN at noise rate 0.20. The red dashed line represents the performance of the state-of-the-art (SOTA).

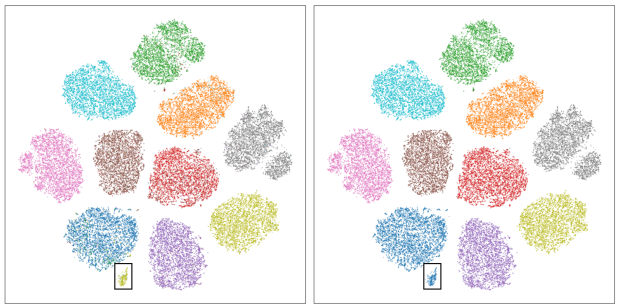
ours is not. As these statistics demonstrate, our method avoids overfitting to given noisy labels and achieves better generalization performance.

5.2. Hyperparameters

To validate the sensitivity of the hyperparameters, experiments are conducted using various values of k_{st} for reverse k-NN and the weight of the structural loss, λ_{st}. The results are shown in Fig. 5.

Figure 5a shows the model's accuracy with respect to different values of the hyperparameter k_{st}. Based on the feature distribution, the model can be trained by smoothing over the correct answer labels of its neighboring samples in a noisy environment. However, if k_{st} is too small, our method cannot fully incorporate the feature distribution. On the contrary, if k_{st} is too large, the number of arrows received from samples of classes other than c will increase, and then structural labels l_{st} are over-smoothed. To avoid over-dilution of structural information, we need to choose an appropriate value of k_{st}. Therefore, we experiment with various values to determine the optimal k_{st}. Our proposed model shows the best performances at a k_{st} of 20.

Figure 5b shows the performance of our proposed method with respect to value of λ_{st}, the weight of the structural loss. When applying structural loss, there is a clear performance improvement compared to the case where it is not applied, λ_{st} is 0. Additionally, it demonstrates stable, improved performance regardless of changes in the value.



(a) (b)

Figure 6. Comparison of t-SNE visualization on CIFAR10 with IDN at noise rate 0.50. (a) Ours on training samples with golden labels. (b) Ours on training samples with predicted labels. There are only smaller amount of wrongly trained samples than DivideMix and SSR.

5.3. The t-SNE visualization

Figure 6 shows t-SNE visualization of features for the proposed method. Unlike DivideMix and SSR in Fig. 1, our method not only learned a relatively simple manifold but also exhibits well-separated clusters, each predominantly representing one class. This confirms that our method demonstrates superior generalization performance.

6. Conclusion

In this paper, we explored the generalization problem in learning with noisy labels (LNL) and proposed a simple yet effective approach to address these challenges. While existing LNL methods, our proposed method employed a reverse k-NN to leverage structural information based on feature distribution. And by learning the structural information, the model had a better feature manifold and robustness against overfitting to noisy labels, thus achieving better generalization performance. The proposed method demonstrated outperformed performance across various datasets including CIFAR10 and CIFAR100 with IDN, CIFAR-N, Animal-10N, Red Mini-Imagenet, Mini-WebVision, and ImageNet.

Acknowledgments. This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-00421, AI Graduate School Support Program, Sungkyunkwan University, and No.RS-2023-00228970, Development of Flexible SW·HW Conjunctive Solution for On-edge Self-supervised Learning).

References

[1] Eric Arazo, Diego Ortego, Paul Albert, Noel O'Connor, and Kevin McGuinness. Unsupervised label noise modeling

and loss correction. In International conference on machine learning, pages 312–321. PMLR, 2019. 1, 2

[2] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. Advances in neural information processing systems, 32, 2019. 2

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. Advances in neural information processing systems, 33:1877–1901, 2020. 1

[4] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. IEEE Transactions on Neural Networks, 20(3):542–542, 2009. 2

[5] Xinlei Chen and Abhinav Gupta. Webly supervised learning of convolutional networks. In Proceedings of the IEEE international conference on computer vision, pages 1431–1439, 2015. 2

[6] Yingyi Chen, Xi Shen, Shell Xu Hu, and Johan AK Suykens. Boosting co-teaching with compression regularization for label noise. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2688–2692, 2021. 6

[7] YunSeok Choi, CheolWon Na, Hyojun Kim, and Jee-Hyong Lee. Readsum: Retrieval-augmented adaptive transformer for source code summarization. IEEE Access, 2023. 1

[8] Filipe R Cordeiro, Ragav Sachdeva, Vasileios Belagiannis, Ian Reid, and Gustavo Carneiro. Longremix: Robust learning with high confidence samples in a noisy label environment. Pattern Recognition, 133:109013, 2023. 7

[9] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. IEEE transactions on information theory, 13(1): 21–27, 1967. 2

[10] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 113–123, 2019. 2, 3, 4

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition, pages 248–255. Ieee, 2009. 2, 4, 5, 7

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018. 1

[13] Mingyu Ding, Bin Xiao, Noel Codella, Ping Luo, Jingdong Wang, and Lu Yuan. Davit: Dual attention vision transformers. In European Conference on Computer Vision, pages 74–92. Springer, 2022. 1

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929, 2020. 1

[15] Chen Feng, Georgios Tzimiropoulos, and Ioannis Patras. Ssr: An efficient and robust framework for learning with unknown label noise. arXiv preprint arXiv:2111.11288, 2021. 1, 2, 4, 5, 6, 7

[16] Evelyn Fix and Joseph Lawson Hodges. Nonparametric discrimination: consistency properties. Randolph Field, Texas, Project, pages 21–49, 1951. 2

[17] Arpit Garg, Cuong Nguyen, Rafael Felix, Thanh-Toan Do, and Gustavo Carneiro. Instance-dependent noisy label learning via graphical modelling. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pages 2288–2298, 2023. 1, 2, 5, 6

[18] Yuan Gong, Yu-An Chung, and James Glass. Ast: Audio spectrogram transformer. arXiv preprint arXiv:2104.01778, 2021. 1

[19] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. Advances in neural information processing systems, 31, 2018. 5, 7

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14, pages 630–645. Springer, 2016. 1, 5

[21] Lihua Hu, Hongkai Liu, Jifu Zhang, and Aiqin Liu. Krdbscan: A density-based clustering algorithm based on reverse nearest neighbor and influence space. Expert Systems with Applications, 186:115763, 2021. 3

[22] Zhizhong Huang, Junping Zhang, and Hongming Shan. Twin contrastive learning with noisy labels. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 11661–11670, 2023. 7

[23] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In International conference on machine learning, pages 2304–2313. PMLR, 2018. 1, 2, 5

[24] Lu Jiang, Di Huang, Mason Liu, and Weilong Yang. Beyond synthetic noise: Deep learning on controlled noisy labels. In International conference on machine learning, pages 4804–4815. PMLR, 2020. 2, 4, 5, 6

[25] Noo-ri Kim and Jee-Hyong Lee. Propagation regularizer for semi-supervised learning with extremely scarce labeled samples. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 14401–14410, 2022. 1

[26] Nikola Konstantinov and Christoph Lampert. Robust learning from untrusted sources. In International conference on machine learning, pages 3488–3498. PMLR, 2019. 2

[27] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 4, 5

[28] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at

scale. International Journal of Computer Vision, 128(7): 1956–1981, 2020. 1

[29] Jin-Seop Lee, Tae-Hyun Kim, Sang-Hwan Jeon, Sung-Hyun Park, Sang-Hi Kim, Eun-Ho Lee, and Jee-Hyong Lee. Automation of trimming die design inspection by zigzag process between ai and cad domains. Engineering Applications of Artificial Intelligence, 127:107283, 2024. 1

[30] Kuang-Huei Lee, Xiaodong He, Lei Zhang, and Linjun Yang. Cleannet: Transfer learning for scalable image classifier training with label noise. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 5447–5456, 2018. 2

[31] Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In International Conference on Learning Representations, 2020. 1, 2, 5, 6, 7

[32] Junnan Li, Caiming Xiong, and Steven CH Hoi. Learning from noisy data with robust representation learning. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 9485–9494, 2021. 7

[33] Shikun Li, Xiaobo Xia, Shiming Ge, and Tongliang Liu. Selective-supervised contrastive learning with noisy labels. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 316–325, 2022. 7

[34] Shikun Li, Xiaobo Xia, Shiming Ge, and Tongliang Liu. Selective-supervised contrastive learning with noisy labels. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 316–325, 2022. 2

[35] Wen Li, Limin Wang, Wei Li, Eirikur Agustsson, and Luc Van Gool. Webvision database: Visual learning and understanding from web data. arXiv preprint arXiv:1708.02862, 2017. 1, 2, 4, 5, 7

[36] Yuncheng Li, Jianchao Yang, Yale Song, Liangliang Cao, Jiebo Luo, and Li-Jia Li. Learning from noisy labels with distillation. In Proceedings of the IEEE international conference on computer vision, pages 1910–1918, 2017. 2

[37] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. Advances in neural information processing systems, 33:20331–20342, 2020. 6, 7

[38] Sheng Liu, Zhihui Zhu, Qing Qu, and Chong You. Robust training under label noise by over-parameterization. In International Conference on Machine Learning, pages 14153–14172. PMLR, 2022. 6

[39] Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. IEEE Transactions on pattern analysis and machine intelligence, 38(3):447–461, 2015. 5

[40] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692, 2019. 1

[41] Michal Lukasik, Srinadh Bhojanapalli, Aditya Menon, and Sanjiv Kumar. Does label smoothing mitigate label noise? In International Conference on Machine Learning, pages 6448–6458. PMLR, 2020. 1

[42] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens Van Der Maaten. Exploring the limits of weakly supervised pretraining. In Proceedings of the European conference on computer vision (ECCV), pages 181–196, 2018. 1

[43] Eran Malach and Shai Shalev-Shwartz. Decoupling" when to update" from" how to update". Advances in neural information processing systems, 30, 2017. 5

[44] Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nathan Srebro. Exploring generalization in deep learning, 2017. 2

[45] Kento Nishi, Yi Ding, Alex Rich, and Tobias Hollerer. Augmentation strategies for learning with noisy labels. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 8022–8031, 2021. 1

[46] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1944–1952, 2017. 5

[47] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in python. Journal of Machine Learning Research, 12(85):2825–2830, 2011. 3

[48] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In International Conference on Machine Learning, pages 28492–28518. PMLR, 2023. 1

[49] Miloš Radovanović, Alexandros Nanopoulos, and Mirjana Ivanović. Reverse nearest neighbors in unsupervised distance-based outlier detection. IEEE transactions on knowledge and data engineering, 27(5):1369–1382, 2014. 3

[50] Nicolae-Catalin Ristea, Radu Tudor Ionescu, and Fahad Shahbaz Khan. Septr: Separable transformer for audio spectrogram processing. arXiv preprint arXiv:2203.09581, 2022. 1

[51] Yanyao Shen and Sujay Sanghavi. Learning with bad training data via iterative trimmed loss minimization. In International Conference on Machine Learning, pages 5739–5748. PMLR, 2019. 2

[52] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. Advances in neural information processing systems, 33:596–608, 2020. 1

[53] Hwanjun Song, Minseok Kim, and Jae-Gil Lee. Selfie: Refurbishing unclean samples for robust deep learning. In International Conference on Machine Learning, pages 5907–5915. PMLR, 2019. 2, 4, 5, 6

[54] Hwanjun Song, Minseok Kim, and Jae-Gil Lee. Selfie: Refurbishing unclean samples for robust deep learning. In

International Conference on Machine Learning, pages 5907–5915. PMLR, 2019. 6

[55] Hwanjun Song, Minseok Kim, and Jae-Gil Lee. Selfie: Refurbishing unclean samples for robust deep learning. In International Conference on Machine Learning, pages 5907–5915. PMLR, 2019. 6

[56] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning, 2016. 6

[57] Yufei Tao, Dimitris Papadias, and Xiang Lian. Reverse knn search in arbitrary dimensionality. In Proceedings of the Very Large Data Bases Conference (VLDB), Toronto, 2004. 2

[58] Sunil Thulasidasan, Tanmoy Bhattacharya, Jeff Bilmes, Gopinath Chennupati, and Jamal Mohd-Yusof. Combating label noise in deep learning using abstention. arXiv preprint arXiv:1905.10964, 2019. 2

[59] Arash Vahdat. Toward robustness against label noise in training deep discriminative neural networks. Advances in neural information processing systems, 30, 2017. 2

[60] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. Journal of machine learning research, 9(11), 2008. 2

[61] Jesper E Van Engelen and Holger H Hoos. A survey on semi-supervised learning. Machine learning, 109(2):373–440, 2020. 2

[62] Andreas Veit, Neil Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge Belongie. Learning from noisy large-scale datasets with minimal supervision. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 839–847, 2017. 2

[63] Jiaheng Wei, Zhaowei Zhu, Hao Cheng, Tongliang Liu, Gang Niu, and Yang Liu. Learning with noisy labels revisited: A study using real-world human annotations. arXiv preprint arXiv:2110.12088, 2021. 2, 4, 5, 6

[64] Pengxiang Wu, Songzhu Zheng, Mayank Goswami, Dimitris Metaxas, and Chao Chen. A topological filter for learning with label noise. In Advances in Neural Information Processing Systems, pages 21382–21393. Curran Associates, Inc., 2020. 2

[65] Zhi-Fan Wu, Tong Wei, Jianwen Jiang, Chaojie Mao, Mingqian Tang, and Yu-Feng Li. Ngc: A unified framework for learning with open-world noisy data. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 62–71, 2021. 7

[66] Xiaobo Xia, Tongliang Liu, Bo Han, Nannan Wang, Mingming Gong, Haifeng Liu, Gang Niu, Dacheng Tao, and Masashi Sugiyama. Part-dependent label noise: Towards instance-dependent label noise. Advances in Neural Information Processing Systems, 33:7597–7610, 2020. 2, 4, 5, 7

[67] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2691–2699, 2015. 2

[68] Youjiang Xu, Linchao Zhu, Lu Jiang, and Yi Yang. Faster meta update strategy for noise-robust deep learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 144–153, 2021. 5, 6

[69] Quanming Yao, Hansi Yang, Bo Han, Gang Niu, and James Tin-Yau Kwok. Searching to exploit memorization effect in learning with noisy labels. In International Conference on Machine Learning, pages 10789–10798. PMLR, 2020. 1

[70] Yu Yao, Tongliang Liu, Mingming Gong, Bo Han, Gang Niu, and Kun Zhang. Instance-dependent label-noise learning under a structural causal model. Advances in Neural Information Processing Systems, 34:4409–4420, 2021. 5, 6

[71] Li Yi, Sheng Liu, Qi She, A Ian McLeod, and Boyu Wang. On learning contrastive representations for learning with noisy labels. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 16682–16691, 2022. 2

[72] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In International Conference on Learning Representations, 2017. 1, 2

[73] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412, 2017. 2, 3, 4, 5, 6

[74] Yikai Zhang, Songzhu Zheng, Pengxiang Wu, Mayank Goswami, and Chao Chen. Learning with feature-dependent label noise: A progressive approach. arXiv preprint arXiv:2103.07756, 2021. 6

[75] Evgenii Zheltonozhskii, Chaim Baskin, Avi Mendelson, Alex M Bronstein, and Or Litany. Contrast to divide: Self-supervised pre-training for learning with noisy labels. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pages 1657–1667, 2022. 1, 2

[76] Xiaojin Jerry Zhu. Semi-supervised learning literature survey. 2005. 2

[77] Zhaowei Zhu, Tongliang Liu, and Yang Liu. A second-order approach to learning with instance-dependent label noise. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10113–10123, 2021. 6