J. Appl. Numer. Optim. 5 (2023), No. 2, pp. 237-253
Available online at http://jano.biemdas.com
https://doi.org/10.23952/jano.5.2023.2.04

IDENTIFYMIX: AN EFFICIENT TWO-STAGE LEARNING APPROACH TO COMBATING LABEL NOISE

KAI TONG[1], XIAO KE[1,2,*]

[1]*Fujian Provincial Key Laboratory of Networking Computing and Intelligent Information Processing, College of Computer and Data Science, Fuzhou University, Fuzhou 350116, China*
[2]*Key Laboratory of Spatial Data Mining & Information Sharing, Ministry of Education, Fuzhou 350003, China*

Abstract. Deep neural networks require correct label annotation during supervised learning. It is inevitable, however, that some labels are noisy during the labeling process. A deep neural network retains incorrect labels during training, resulting in a degradation of performance. Therefore, it is essential to identify samples with potentially correct labels. In state-of-the-art methods, small-loss samples are chosen for subsequent training through a sample selection strategy. Howerver, it typically ignores the imbalance in noise ratios between mini-batches when performing sample selection within each mini-batch. Further, numerous valuable samples with high losses are discarded, which adversely affects the generalization performance of the model, particularly under conditions of high noise ratios. To this end, this paper proposes IdentifyMix, an effective two-stage learning approach for noisy robust learning that combines an unique sample selection strategy and the semi-supervised learning technique. By observing how the dynamics of network training are changing, AUM (Area Under the Margin) provides a criterion that is applied in this research to identify mislabeled data. Moreover, by combining semi-supervised learning with contrastive learning and data augmentation, it is possible to extract more useful information from mislabeled samples. Experiments on several synthetic and real-world noise benchmarks demonstrate the effectiveness of IdentifyMix compared with state-of-the-art methods.

Keywords. Image classification; Contrastive learning; Noisy labels; Semi-supervised learning; Sampling strategy.

2020 Mathematics Subject Classification. 68T05, 68Q32.

1. INTRODUCTION

Deep neural networks (DNNs) led to substantial improvements in many computational version tasks [1, 2]. However, the quality of the labels is an important determinant of the performance of deep neural networks. Annotating large amounts of data correctly involves high labour costs, expert knowledge, and a considerable amount of time. Human annotation of a large amount of data with correct labels is inevitable to produce noisy labels. According to a recent study [3], as a result of overfitting and performance reduction, DNNs tend to memorize

*Corresponding author.
E-mail addresses: kex@fzu.edu.cn (X. Ke), tongk0630@foxmail.com (K. Tong).
Received December 25, 2022; Accepted June 16, 2023.

noisy data during training. The challenge of learning noisy robustness for deep neural networks is therefore considerable.

Noisy label comprises different noisy distributions in image classification tasks [4, 5, 6]. In-distribution noise types were composed of samples with incorrect labels, but the content of samples is part of the classes of datasets. It was usually associated with either a symmetric or asymmetric random distribution of noise when synthetic in-distribution noise was introduced. The former indicates label flips to classes with a uniform probability, and the latter suggests label flips to classes with a unique probability. The real-world label noise types were usually out-of-distribution [5, 7], in which the content of images does not belong to the class of datasets.

Many recent studies have focused on learning noisy robustness recently; see, e.g., [6, 8, 9, 10] and the references therein. In early approaches, losses are primarily corrected during training. There are some methods that correct losses by introducing a noise transition matrix [11, 12]. Nevertheless, estimating the noise transition matrix is difficult, requiring either prior knowledge or a subset of data that has been labeled. According to some methods, a noise robust loss function could be designed that corrects losses according to predictions derived from DNNs [13, 14]. The disadvantage of these methods is that they are prone to failure when the noise ratio is high.

DNNs would also be trained with reweighted [15] or selected training samples as an effective means of mitigating the negative effects of noisy labels [6, 8, 10]. To identify clean samples, it is necessary and important to develop a proper criterion. Several studies have demonstrated that DNNs tend to learn clean and simple patterns before overfitting noisy labels; see, e.g., [5, 8, 16, 17] and the references therein. Therefore, state-of-the-art methods (e.g., Co-teaching [8], Co-teaching+ [17], and JoCoR [16]) attempt to select a human-defined percentage of samples with small losses as clean ones. Even though the small-loss sample selection strategy has demonstrated promising performance gains, these methods assume that all mini-batches have identical noise ratios. Thus, they select samples within each mini-batch based on an estimated noise rate. Nevertheless, this assumption may not apply in real-world situations (e.g., Animal-10N [18]), and estimating the noise rate accurately is also challenging. Comparison with the above methods, AUM [19] adopts an unique sampling strategy that allows it to distinguish clean samples from contaminated samples by observing the difference between correctly labeled samples and incorrectly labeled samples during dynamic model training. It has the advantage of successfully overcoming the inability of the above methods based on the small-loss strategy to sample effectively in real scenarios, as well as the reduced capability of distinguishing noisy labeled samples from clean labeled samples in high noise rate scenarios due to the insufficient number of clean samples taken between mini-batches. However, it should be noted that the AUM sampling strategy with a weighted method may result in underfitting if the dataset has a large number of its own categories and is contaminated with moderate or high noise rates.

Motivated by the semi-supervised learning (SSL) technique [20, 21], a simple yet effective approach is presented in this paper named IdentifyMix, a novel two-stage learning approach that combines an innovative sample selection mechanism with semi-supervised learning to address aforementioned issues. Specifically, an AUM sampling strategy distinguishes clean samples from noisy datasets in the first place, which is intended to enhance the AUM sampling mechanism and enhance the capability of the model to distinguish between labeled types of data by implementing a threshold sampling strategy that would replace common cross-validation.

Additionally, a semi-supervised learning approach is then applied to take full advantage of the large amount of useful information available from samples with incorrect labels to address the problem of underfitting of the model to clean samples. Ultimately, contrastive learning with Mixup [22] data augmentation allows the model to learn from sample features with limited correct labels efficiently. There is no doubt that large deep neural networks are powerful, but they also suffer from some shortcomings, such as memory and sensitivity to noisy examples. Therefore, Mixup was proposed to relieve these problems. As a summary, the main contributions are as follows:

- An innovative noise detection strategy distinguishes clean samples from noisy datasets. The Area Under the Margin (AUM) statistic is employed as a criterion without applying the small-loss measure, which relies on differences in training dynamics between clean and mislabeled samples.
- In the SSL phase, the risk of noisy label memorization is minimized by performing supervised feature learning using contrastive loss and applying Mixup to clean and labeled samples to improve the performance of the model in real-world, synthetic noisy datasets at various noise ratios.
- Experimental results demonstrate that IdentifyMix achieves significant performance improvements over state-of-the-art methods on multiple benchmarks with different noise ratios. In addition to ablation studies, qualitative results were also provided to examine the effect of different components.

2. Related Works

There have been several proposals for alleviating label noise. A summary of some recent approaches to noise-robust learning and contrastive learning methods was provided.

Designing noise-robust loss. The mainly employed Cross Entropy(CE) loss is confirmed by overfitting when there is noise in the label. Therefore, studies have been devoted to designing novel loss functions that tolerate noise labels. NLNL [23] developed an novel loss, a form of selective negative and positive learning for robust noisy learning. JNPL [24], an improvement of NLNL, unified the filtering pipeline into a single stage instead of a three-stage pipeline. In contrast to NLNL, JNPL [24] implemented a three-stage filtering pipeline in a single stage. APL [13] built new loss functions(NCE+RCE, NFL+RCE) with theoretically guaranteed robustness and sufficient learning properties to address the existing robust loss functions suffering from an underfitting problem. One of the most representative methods is APL, which is capable of combining two distinct existing noise robustness loss functions to enhance the noise robustness of the network through parameter settings compared to the previous methods. Nevertheless, most noise robustness loss functions still require performance enhancements in many classes and in high noise cases.

Noisy samples refusion. Co-teaching [8] utilized small-loss data of one network to teach its peer network for the further training in each mini-batch. Co-teaching+ [17] first predicted each min-batch with two networks but uses disagreement of samples only to compute the training loss. JoCoR [16] trained the two networks as a whole with a joint loss of weight goals: making the two predictions argree with each other, and making the predictions stick to ground-true labels as far as possible. MentorMix [6] provided a simple but effective method to overcome both

synthetic and real noisy labels. Co-learning [5] trained a single shared encoder network with two heads (the self-supervised and noisily-supervised) that constrain each other and maximizing the agreement between them in the latent space. The main advantage of these proposed methods mentioned above is that they can be applied to noise-robust learning by selecting some correctly labelled samples based on a small-loss criterion and increasing their contribution while reducing the contribution of mislabeled samples. The disadvantage of these methods is that they are unable to leverage mislabeled samples effectively in low-noise scenarios.

Selecting clean labels. Some studies [12, 19, 25] make contribute to seperating clean labels from noisy dataset. AUM [19] was committed to automatically identifying, subsequently removing mislabeled samples from the training datasets and mitigating their impact when training networks. INCV [25] was dedicated to applying cross-validation to randomly split noisy datasets and iteratively filter training data, identifying the most samples with correct labels. Then, adopting the Co-teaching [8] strategy to identify clean samples further. BMM [26] proposed a beta mixture to estimate this probability and correct the loss by relying on the network prediction. These approaches provide some advantages over dynamically and efficiently identifying potentially clean labeled samples and separating noisy labeled samples during the training process for a limited number of classes. Nevertheless, they often result in class imbalance and fail to produce satisfactory results for a significant number of classes.

Noisy label rejection. Recently, several approaches have performed semi-supervised learning by treating mislabeled samples as unlabeled samples, which could effectively exploit the content of samples and refuse their noisy labels. ELR [27] leveraged a semi-supervised learning technique to produce target probabilities relying on the model outputs and designed a regularization term that leads toward these targets, implicitly preventing memorization of incorrect labels. MOIT [28] proposed a technique related to joint semi-supervised and contrastive learning. UNICON [29] proposed a noval sample selection mechanism which detects clean-noisy samples and applies SSL to iterative training. An essential advantage of these approaches is that they combine sample selection mechanisms with semi-supervised learning, which is motivated to train the network and improve the model's generalization performance. As a result, they achieve impressive results in high-noise scenarios as opposed to the previous methods.

Contrastive representation learning. In the recent results on self-supervised learning [30, 31, 32] contrastive similarity learning frameworks have been demonstrated to be effective in the learning of representations. Maximizing (minimizing) similarity is the common denominator among these methods between positive (negative) pairs. MOIT [28] incorporated contrastive learning with classification to improve label noise performance. Co-learning [5] involved both supervised and self-supervised learning cooperatively. CLIM [33] demonstrated that initializing robust supervised methods using representations learned through contrastive learning results in enhanced performance under label noise. By applying contrast feature learning to improve the performance of the model under a dataset with noisy labels is one of the most effective methods mentioned above. However, due to the need for positive and negative pairs required for contrastive learning, a significant amount of memory is required during the training process, which consumes a significant amount of resources.

3. METHODOLOGY

In this section, we present IdentifyMix, an efficient two-stage learning method, for robust learning from noisy datasets. An overall of the method is demonstrated in Figure 1. The overall framework consists of two critical components, namely, an innovative sampling method and SSL-training. In the first stage, a customized sample selection strategy is developed employing the AUM criteria, as demonstrated in Figure 1a, in which one network separates the noisy training dataset into a clean labeled set $X(D_{clean})$ and a noisy unlabeled set $U(D_{noisy})$. During the second stage, the other network incorporates both labeled and unlabeled samples for semi-supervised learning. In their respective periods, the two networks are independent of one another and executed sequentially. Algorithm 1 describes the complete algorithm.

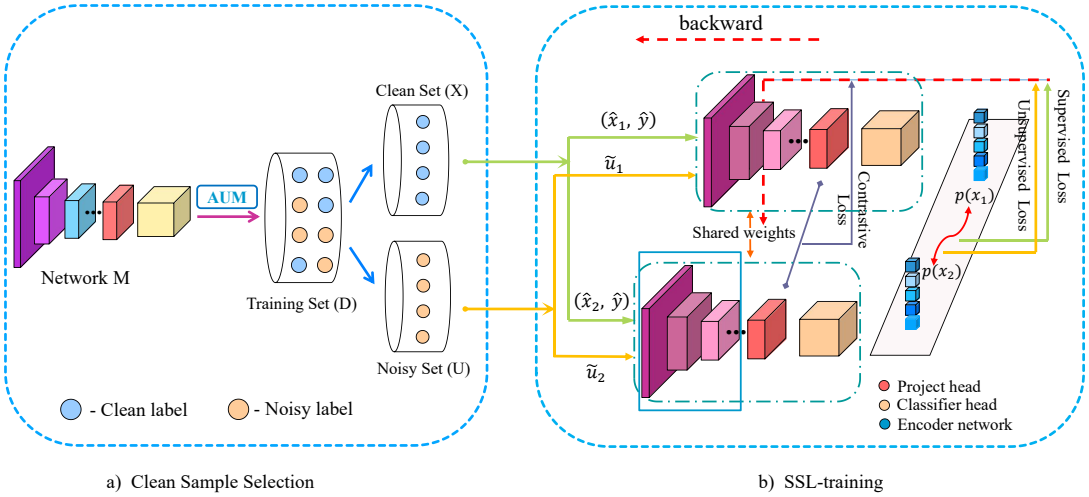

FIGURE 1. IdentifyMix consists of two modules: a) clean sample selection and b) SSL-Training. IdentifyMix trains two networks separately in sequence. For a) it is calculated that network M computes the *aum* values of each sample to separate the dataset into two training subsets, namely the labeled dataset (mostly clean) and the unlabeled dataset (mostly noisy), on the basis of which the other network is trained. For b) the other network is designed to perform semi-supervised learning with a Mixup and contrastive loss on labeled data.

3.1. **Identifying Mislabeled Data.**

Assuming training dataset $D_{train} = \{x_i, y_i\}_{i=1}^{N}$ comprises two data types. A mislabeled sample is one where a sample does not match the assigned label. A correct-labeled sample has an allocated label matching the ground truth of the sample. Some correctly labeled examples might be easy to learn if they are common. Others might be hard to learn if they are rare occurrences. Assuming both easy and hard correctly-labeled samples in D_{train} optimizes model generalization even through mislabeled samples hurt generalization. The goal of the sample selection strategy is to identify noisy data in D_{train} simply by observing differences in training dynamics among data.

3.1.1. *Area Under the Margin (AUM) Ranking.* Let $(x, y) \subseteq D_{train}$ be a sample, and let $z^{(t)} \subseteq R^c$ be its logits vector at epoch t. The margin at epoch t indicates how much difference there is

between the assigned logit and other logits which are not assigned:

$$M^{(t)}(x,y) = z_y^{(t)}(x) - Max_{i \neq y} z_i^{(t)}(x),$$

where $z^{(t)}$ is also a pre-softmax output at epoch t, i corresponds to class i, $M^{(t)}$ means the margin at epoch t, and x denotes the sample and y is the corresponding label.

The negative margin corresponds to an incorrect prediction, whereas the positive margin corresponds to a confidently correct prediction. The sample with an assigned label could be considered incorrect if gradient updates from similar samples are compared to the sample with a considerable negative margin. As stated in the hypothesis, it was desirable that a mislabeled sample had a small margin or a negative margin when compared to a correct labeled sample. Consequently, the assumption above could be satisfied by averaging the margin of the sample measured at each training epoch. A metric is calculated to demonstrate the above assumption, according to (3.2):

$$aum(x,y) = \frac{1}{T} \sum_{t=1}^{T} M^{(t)}(x,y), \tag{3.1}$$

where T is the total number of training epochs, and $aum(x,y)$ denotes the weighted average of the margins measured by sample x in each training epoch t. AUM refers to the area consisting of the margins of sample x measured at each training epoch t, and is an intuitive presentation of the *aum* training process.

This metric illustrates the logits for different Cifar-10N training samples over time during the training of Resnet-32 demonstrated in Figure 2. As demonstrated in Figure 2a, the clean automobile samples are easy to learn, while the clean automobile samples in Figure 2b are more challenging. Figure 2a demonstrates the examples of cleanly labeled automobiles that are easy to distinguish. It is difficult to identify the clean-labeled car samples in Figure 2b owing to their blurry edges. AUM is assessed by comparing the automobile logical values with the largest other logical values during the training process, and a larger area indicates that the model is better able to distinguish between well-learned samples. Both in Figure 2a and Figure 2b, the logical values for automobiles are much larger than those for other logical values. In general, automobile logical values are higher than other logical values. Since the edges of the samples in Figure 2b are blurred, the blue area(AUM) in Figure 2b becomes smaller during training compared to the blue area in Figure 2a. Figure 2c demonstrates that the logical values of the mislabeled automobile samples are negative and much smaller than the other logical values.

3.1.2. *Creating threshold samples.* As a complement to the AUM sample sampling strategy, the threshold sample set are recommended as an alternative to the cross-validation set to enhance the capacity of the model to distinguish noisy samples. The training process relies on threshold samples in order to mimic the training dynamics associated with mislabeled data. As demonstrated in Figure 2d, threshold samples are assigned alternate labels during dynamic training in order to simulate mislabeled samples. This procedure is designed to ensure that the model could effectively distinguish between clean and incorrectly labeled samples. There is a possibility of mislabeled data with *aum* values similar to or worse than threshold samples. Thus threshold samples derived from a subset of training data are constructed by reassigning labels to nonexistent classes.

For example, consider that the training set consists of N samples that belong to C categories. Assign labels to $C+1$ at random for $N/(C+1)$ samples that are selected as threshold samples

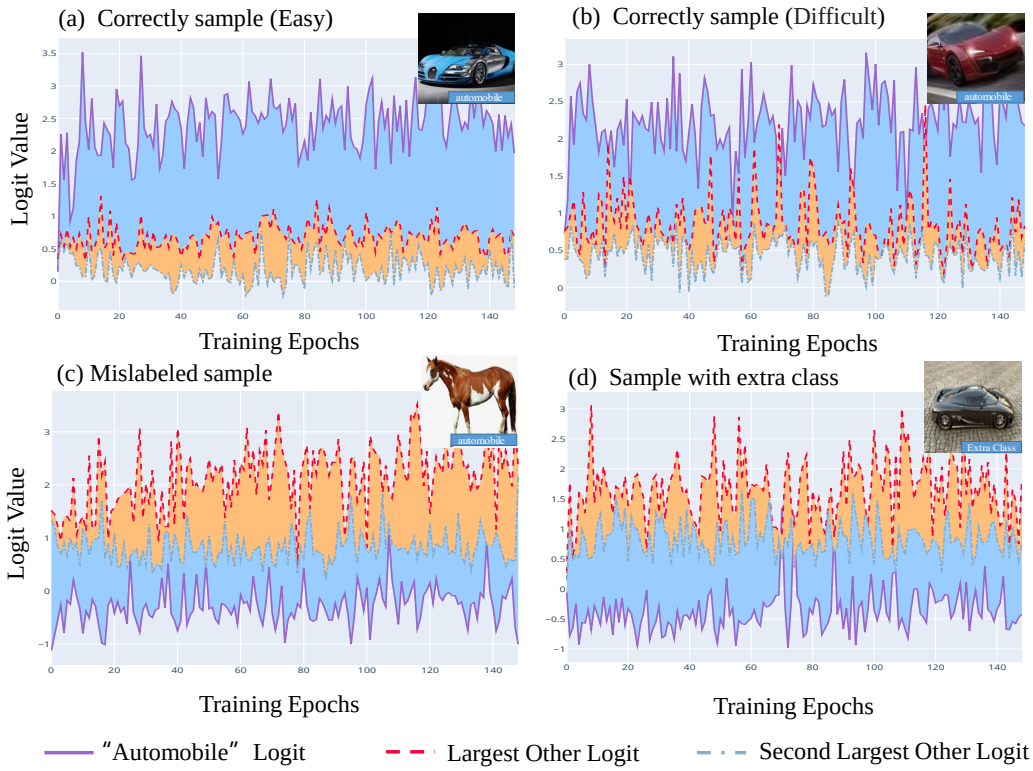

FIGURE 2. An analysis of the Area Under the Margin (AUM) metric. An illustration of logit values for easy-to-learn automobiles (a), hard-to-learn automobiles (b), horses mislabeled as automobiles (c) and automobiles with reassigned labels (d). The AUM corresponds to the shaded region between the Automobile logit and the Largest other logit. Purple and red lines indicate the *aum* value for the sample at epoch t, these values are positive and negative, respectively. The *aum* values of correctly labeled samples are larger than those of incorrectly labeled samples.

to ensure that the extra class is on average as likely as other classes. Since threshold samples are only capable of raising the assigned $C+1$ logits through memorization, threshold samples are expected to have a small and likely negative margin.

There are two advantages for a threshold sample with an additional class $C+1$. First, all threshold samples are guaranteed to be simulated as mislabeled samples. However, the random assignment of labels from 1 to C could result in mistakes, which would defeat the purpose of the initial design. Moreover, additional $C+1$ classification tasks do not affect the main classification tasks. Data with lower *aum* values than the 99th percentile threshold samples are identified as mislabeled samples for Cifar-10N and Cifar-100N. With respect to Anmial-10N, this paper identifies data with *aum* values less than the αth percentile threshold samples. Positive logit values indicate correct samples, while negative logit values represent noisy samples. During the experiment, the hyperparameters α and β were manipulated. It should be noted, however, that the sample selection mechanism and weighting method can result in underfitting of the model when there are high levels of noise and multiple classes in the dataset. Therefore, the introduction of semi-supervised learning was intended to address this issue.

Algorithm 1: IdentifyMix. Line 1-10: sample selection; Line 11-28: SSL training;

Input: θ_1, θ_2, training dataset (X,Y), MaxEpoch M_1, MaxEpoch M_2, noise type NT, margin metric strategy Aum, project_head $P(;)$, classfier_head $G(;)$, encoder network $F(;)$, positive clean samples strategy $Pos\_s(;)$, negative clean samples strategy $Neg\_s(;)$, contrastive learning with supervision $Supcon(;)$, data augument $Mixup$, cross-entropy $H(;)$, certainty threshold ζ, unsupervised loss weight λ_u, contrastive loss weight λ_{cl}

Output: $D_{clean}(X,Y)$, $D_{noise}(X,Y)$, θ_2

Construct a modified training set $D^{'}_{train}$ which includes D_{thr}
$D^{'}_{train} = \{(x,c+1) : x \in D_{thr}\} \cup (D_{train} \backslash D_{thr})$

Train a network on $D^{'}_{train}$ until the first learning rate drop, measuring the AUM of all data.

if *NT is symmetric or asymmetric* **then**
    Compute $\gamma \leftarrow 99\%$ of aums of threshold samples
    Identify mislabeled samples using γ:
    $D_{noise}(x,y) = \{(x,y) \in (D_{train} \backslash D_{thr}), aums(x,y) \leq \gamma\}$
    Get correct samples from difficult but benefit samples:
    $D_{clean}(x,y) = D^{'}_{train} - D_{noise}(x,y)$
else if *NT is real world noise* **then**
    $D_{clean}(x,y) = Pos\_s(x,y) + Neg\_s(x,y)$
    $D_{noise}(x,y) = D^{'}_{train}(x,y) - D_{clean}(x,y)$
end
Set $D_{clean}(X,Y)$, $D_{noise}(X,Y)$ as $D_{labeled}(X,Y)$, $D_{unlabeled}(X,Y)$
while $e \leq M_2$ **do**
    $X_e = \{(x_i,y_i) \in D_{clean}(X,Y)\}$, $U_e = \{u_i \in D_{noise}(X,Y)\}$
    for $t = 1$ **to** *num_iters* **do**
        $\hat{x}_{t,m} = weak\_Augment(x_t)(m = 1,2)$
        $\hat{u}_{t,1} = weak\_Augment(u_t), \quad \hat{u}_{t,2} = strong\_Augment(u_t)$
        $\hat{f}_{t,1} \leftarrow F(\hat{x}_{t,1};\theta_2), \quad z_{t,1} \leftarrow G(\hat{f}_{t,1};\theta_2)$
        $\hat{f}_{t,2} \leftarrow F(\hat{x}_{t,2};\theta_2), \quad z_{t,2} \leftarrow G(\hat{f}_{t,2};\theta_2)$
        $v_{t,1} \leftarrow P(\hat{f}_{t,1};\theta_2), \quad v_{t,2} \leftarrow P(\hat{f}_{t,2};\theta_2)$
        $X_t = cat([v_{t,1}, v_{t,2}], dim = 0), \quad Y_t = cat([y_t, y_t], dim = 0)$
        $L_{cl} = Supcon(X_t, Y_t), \quad L_x = Mixup(z_{t,1}, z^{'}_{t,1})$
        $\hat{f}_{t,1} \leftarrow F(\hat{x}_{u,1};\theta_2), \quad \hat{z}_{t,1} \leftarrow G(\hat{f}_{t,1};\theta_2)$
        $\hat{f}_{t,2} \leftarrow F(\hat{x}_{u,2};\theta_2), \quad \hat{z}_{t,2} \leftarrow G(\hat{f}_{u,2};\theta_2)$
        $L_u = \frac{1}{\mu B} \sum_{b=1}^{\mu B} H(\hat{z}_{t,2}, \hat{p}_u)\{max(p(\hat{z}_{t,1};\theta_2)) \geq \zeta\}$
        $L = L_x + \lambda_u L_u + \lambda_{cl} L_{cl}$
        $\theta_2 \leftarrow SGD(L, \theta_2)$
    end
end

3.2. **SSL-Training.** Figure 1b demonstrates the details of SSL-Training with contrastive learning and Mixup. FixMatch [34] was exploited for the SSL. It generates two copies of each sample with weak and strong augmentations [35] from D_{noise}. Additionally, it would generate two views of each sample with two random weak augmentations [35] from D_{clean}. Mixup [22] data augmentation is also applied between two different copies of each mini-batch with weak augmentation from D_{clean}. However, feature learning in such a SSL manner still suffers from noise threats for memorization. As a consequence of the presence of noisy samples in the clean set, DNNs memorize a certain portion of noisy samples during training.

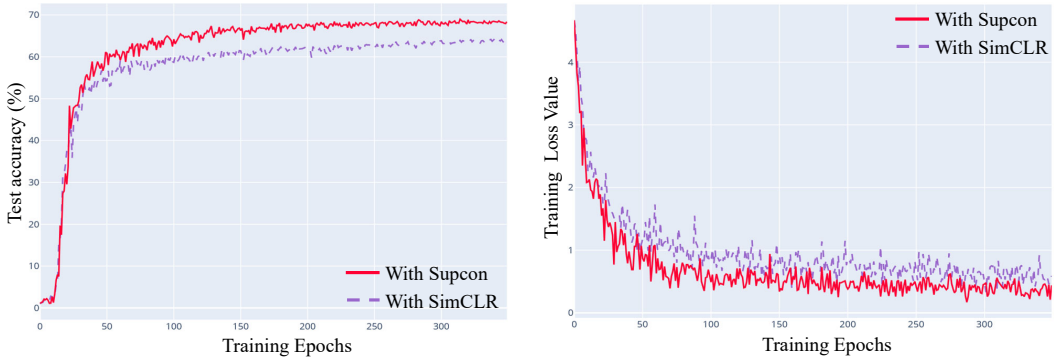

FIGURE 3. Analyze the test accuracy and training loss of IdentifyMix in comparison with Supcon and SimCLR on Cifar-100N with 60% symmetric noise.

3.2.1. *Contrastive learning under supervision.* To address the above issue, the SSL pipeline incorporates contrastive representation learning [30, 31, 36] to promote feature learning under clean labels. SimCLR [30] learns representations by maximizing agreement between different augmented views of the same data example through contrastive loss in the latent space. Under fully supervised conditions, Supcon [31] extends the self-supervised contrastive approach by leveraging label information. As demonstrated in Figure 3, Supcon produced higher test accuracy than SimCLR in the experimentation process. In the training process, it tends to converge faster and the lower bound of convergence increases. Assume that the training minibatch $\{(x_i, y_i)\}_{i=1}^{2N}$ of image-label pairs x_i and y_i contains 2N images. Images are mapped to low-dimensional representations z_i by learning encoder networks F_θ and projection networks H_ϑ with parameters θ and ϑ. Specifically, an intermediate embedding $v_i = F_\theta(x_i)$ is generated and subsequently transformed into the representation $w_i = H_\vartheta(v_i)$. Finally, $z_i = w_i/\|w_i\|_2$ is the L_2-normalized low-dimensional representation used to learn based on the per-sample loss. In accordance with (3.3):

$$L_{cl} = L_i(z_i, y_i) = \frac{1}{2N_{y_i} - 1} \sum_{j=1}^{2N} \mathbb{I}_{i \neq j} \mathbb{I}_{i=j} P_{i,j} \tag{3.2}$$

and

$$P_{i,j} = -\log \frac{exp(z_i \cdot z_j/\tau)}{\sum_{k=1}^{2N} \Gamma_{k \neq i} exp(z_i \cdot z_k/\tau)}, \tag{3.3}$$

where $P_{i,j}$ denotes the j-th component of the temperature τ scaled softmax distribution of inner products $z_i \cdot z_j$ of representations from the positive pair of samples x_i and x_j, which can be

interpreted as a probability, $\Gamma_{k\neq i} \subseteq \{0,1\}$ is an indicator function estimating to 1 iff $k \neq i$, and $P_{i,j}$ is summarized in (3.4) across all $2N_{y_i}$ samples x_j in the minibatch sharing label with $x_i(y_i = y_j)$ expect for the self-contrast case $(i = j)$. As defined by the indicator function $\mathbb{I}_B \subseteq \{0,1\}$ that returns 1 when condition B is satisfied and 0 otherwise. Minimizing L_i means restructing F_θ and H_ϑ to put together the feature representations z_i, z_j when they share the same labels $(y_i = y_j)$, while pushing them apart when they do not.

3.2.2. *Data augmentation: Mixup.* As a means of mitigating the negative effects of noisy labels, Mixup [22] data augmentation was introduced for semi-supervised learning from D_{clean} to acquire labeled training subsets after each mini-batch data mixture augmentation $X^{'}$. Concretely, for a pair of mini-batch samples (x_a, x_b) on $X^{'}$ and their corresponding labels (y_a, y_b), the mixed $(x_i^{'}, y_i^{'})$ is computed by:

$$x_i^{'} = \lambda x_a + (1-\lambda)x_b,\ y_i^{'} = \lambda y_a + (1-\lambda)y_b,$$

where $\lambda \subseteq [0,1] \sim Beta(\alpha, \alpha)$.

The loss function for FixMatch consists of two cross-entropy loss terms: a supervised loss L_χ and an unsupervised loss L_u. A supervised loss L_χ, as demonstrated in Eq. (3.7), applied clean and labeled samples with Mixup on $X^{'}$:

$$L_\chi = -\frac{1}{|X'|}\sum_{x,y\subseteq X'}\sum_c y_c log(P_m^c(x;\theta)), \tag{3.4}$$

where $P_m^c(;)$ represents the model's predicted class distribution. θ is the parameters of the model.

An unsupervised loss L_u is displayed in Eq. (3.8), which efficiently predicts the classes of a large subset of unlabeled samples on the basis of their own representations. FixMatch [34] computes an artificial label for each unlabeled and noisy sample on $U^{'}$, which is then exploited in standard cross-entropy loss. According to Eq. (3.9), an artificial label was obtained by computing the model's predicted class distribution from a weakly-augmented version of an unlabeled sample u from $U^{'}$. Thus Eq. (3.10) would be a pseudo-label for a strongly-augmented version of u

$$L_u = -\frac{1}{|U'|}\sum_{u\subseteq U'}\sum_c \mathbb{A}(max(q_b) \geqq \varepsilon)\hat{q}_b log(P_m^c(A(u);\theta)), \tag{3.5}$$

$$q_b = p_m(\alpha(u);\theta), \tag{3.6}$$

and

$$\hat{q}_b = \arg\max(p_m(\alpha(u);\theta)), \tag{3.7}$$

where ε is a scalar hyperparameter meaning preserving pseudo labels for unlabeled samples when fulfilling the threshold above and $\mathbb{A} \subseteq \{0,1\}$ is an indicator function evaluating to 1 when conditions are satisfied in Eq. (3.8).

As a result, the total loss is as follows: $L = L_\chi + \lambda_u L_u + \lambda_{cl} L_{cl}$. In experiments, setting λ_u as 1 and λ_{cl} as 2.5×10^{-2} and utilizing λ_u to control the strength of the unsupervised loss and λ_{cl} to influence the strength of the contrastive loss.

4. NUMERICAL EXPERIMENTS

In this section, comprehensive experiments are used to verify the effectiveness of the method proposed in this paper.

4.1. **Datasets and Implementation Details.**

Simulated noisy datasets. A proposed algorithm in this paper is verified for feasibility on two simulated noisy datasets, *i.e.*, Cifar-10N [29] and Cifar-100N [29]. Both Cifar-10N and Cifar-100N contain 50K training images and 10k test images of size $32 \times 32 \times 3$. Following previous works [28, 37, 38], two types of label noise are discussed: symmetric and asymmetric noise. Symmetric noise is derived by randomly replacing the labels with all possible labels for a certain percentage of the training data. In asymmetric noise, labels flip to incorrect classes as a result of label flipping. (e.g., truck $\rightarrow$ automobile, bird $\rightarrow$ airplane).

For Cifar-10N/100N datasets, the first stage employs the Resnet-32 model and trains it through SGD optimization with a momentum of 0.9, a weight decay of 10^{-4}, and a batch size of 64. The network is trained for 150 epochs and the warm-up training has 10 epochs. The learning rate was initialized at 0.1. In the second stage, a wide Resnet convolutional neural network is applied as the backbone network. Its width is set to 4 and depth to 28. The network is trained for 350 epochs and the warm-up training has 20 epochs. Training it with SGD optimization with a momentum of 0.9, a 5×10^{-4} weight decay and a batch size of 128. Setting the initial learning rate as 3×10^{-2}, and reducing it by a factor of 10 after 150 and 500 epochs. All images are resized to 32×32. We alwagys use the Mixup parameter $\alpha = 1$ and loss weights $\lambda_u = 1, \lambda_{cl} = 0.025$.

Real-world noisy dataset. A proposed algorithm in this paper is evaluated for effectiveness on a real-world noisy dataset. Animal-10N is a real scene noise dataset with manual annotation for online images, which contains ten confusion classes with 50k image samples of size $64 \times 64 \times 3$ in the training set and 5k image samples of size $64 \times 64 \times 3$ in the test set. Noise labels are imported due to manual labeling errors, and the noise ratio accounts for approximately 8% of the dataset.

For Animal-10N dataset, the first stage employs the Resnet-50 model and trains it through SGD optimization with a momentum of 0.9, an initial learning rate of 0.1, a 5×10^{-4} weight decay and a batch size of 64. The model is trained for 150 epochs and warm-ip training has 10 epochs in the first stage. In the second stage, a wide Resnet convolutional neural network is applied as the backbone network. Its width is set to 6 and depth to 28. The network is trained for 350 epochs and the warm-up training has 30 epochs. Training it with SGD optimization with a momentum of 0.9, a 5×10^{-4} weight decay and a batch size of 64. Setting the initial learning rate as 3×10^{-2}, and reducing it by a factor of 20 after 150 and 500 epochs. All images are resized to 64×64. We always use the Mixup parameter $\alpha = 1$ and loss weights $\lambda_u = 1, \lambda_{cl} = 0.025$.

4.2. **Comparison with state-of-the-art methods.**

In this section, we present empirical comparisons between SOTA approaches from five different perspectives, including noise-robust loss methods (NFL+RCE [13], NLNL [23], and NCE+RCE [13]), noisy samples refusion approaches (Co-learning [5], JoCoR [16], and MentorMix [6]), selecting clean label strategies (AUM [19], INCV [25], and BMM [26]), noisy label rejection techniques (ELR [27], JNPL

[24], and MOIT [28]), and contrastive learning (Co-learning [5], MOIT [28], and CLIM [33]). Note that we directly report their experimental results from related papers.

Results on Simulated Noisy Datasets. As demonstrated in Table 1 and Table 2, for the method IdentifyMix proposed in this paper, it achieves competitive performance with symmetric noise compared with other recent state-of-the-art algorithms in most cases, *e.g.*, MOIT and CLIM. Specifically, our algorithm outperforms the CLIM in most cases of symmetric noise when the noise rate lies between 20% and 60%. However, when the noise rate reaches 80%, our algorithm forfeits its performance advantage compared with the CLIM. There is approximately a 3%/12% performance loss for Cifar-10N/Cifar-100N. It is possible that the CLIM initializes the data in a supervised scenario by contrasting feature learning so as to minimize the drop in the number of cleanly labeled samples at high noise rates. In contrast, our algorithm employs semi-supervised learning, while the number of clean samples obtained is limited when the noise rate is 80%, resulting in underfitting of the model and reduced performance. For asymmetric noise, IdentifyMix consistently achieves the best performance when noise ratio is 20%. Howerver, MOIT could perform better when nosie ratio is 40%. The experimental results demonstrate the effectiveness of our algorithm in most noisy scenarios.

TABLE 1. Comparison with state-of-the-art methods in test accuracy(%) on Cifar-10N and Cifar-100N with symmetric noise. Note that the best results are marked in **bold**.

Dataset	Cifar-10N				Cifar-100N			
Methods/Noise ratio	20%	40%	60%	80%	20%	40%	60%	80%
Cross Entropy	83.95	67.58	43.55	17.32	57.32	45.64	24.30	8.06
AUM [19]	90.20	87.50	82.10	54.40	65.50	61.30	53.00	31.70
INCV [25]	89.50	86.80	81.10	53.30	58.60	55.40	43.70	23.70
BMM [26]	94.00	92.80	90.30	74.10	73.70	70.10	59.50	39.50
NLNL [23]	73.70	63.90	50.68	29.53	46.99	30.29	16.60	11.01
NCE+RCE [13]	90.25	86.81	79.92	57.06	65.31	58.67	46.82	27.42
NFL+RCE [13]	89.14	86.05	79.78	55.06	65.31	59.48	47.12	25.80
JoCoR [16]	91.84	88.15	59.2	20.72	71.75	63.96	37.84	7.32
MentorMix [6]	95.60	94.20	91.30	81.00	78.60	71.30	64.60	41.20
Co-learning [5]	92.52	90.49	80.30	62.49	66.78	55.03	49.38	36.12
ELR [27]	92.12	91.43	88.87	80.69	74.68	68.43	60.05	30.27
MOIT [28]	92.88	90.55	85.02	70.53	72.78	67.36	60.13	45.63
JNPL [24]	93.53	91.89	88.45	35.65	70.94	68.11	61.26	17.55
CLIM [33]	94.07	93.75	91.02	**90.56**	77.22	69.87	68.54	**60.5**
IdentifyMix(Ours)	**95.88**	**95.13**	**91.35**	87.64	**78.67**	**76.37**	**70.62**	48.44

Results on the Real-World Noisy Dataset. The method IdentifyMix is validated on the real-world dataset Animal-10N [15], which contains noisily-labeled images collected from Web.

As demonstrated in Table 3, IdentifyMix achieves the superior results than other state-of-the-art methods, including three recent methods that also utilize contrastive learning. The results suggest that our method is able to handle realistic scenes.

TABLE 2. Comparison with state-of-the-art methods in test accuracy(%) on Cifar-10N and Cifar-100N with asymmetric noise. Note that the best results are marked in **bold**.

Dataset	Cifar-10N		Cifar-100N	
Methods/Noise ratio	20%	40%	20%	40%
Cross Entropy	87.67	76.37	62.12	44.55
AUM [19]	89.70	58.70	59.70	40.20
INCV [25]	88.30	79.80	56.80	44.40
BMM [26]	86.56	74.28	69.12	46.97
NLNL [23]	93.35	89.86	63.12	45.70
NCE+RCE [13]	88.56	79.59	62.68	46.79
NFL+RCE [13]	88.73	79.27	63.12	42.97
JoCoR [16]	91.19	83.61	65.05	45.14
MentorMix [6]	91.36	89.19	72.32	60.61
Co-learning [5]	91.07	81.42	65.26	47.62
ELR [27]	93.31	85.34	74.88	70.00
MOIT [28]	93.19	**92.27**	73.34	**71.55**
JNPL [24]	93.45	90.72	69.95	59.51
CLIM [33]	93.54	90.27	71.26	59.26
IdentifyMix(Ours)	**95.09**	89.69	**76.82**	62.55

4.3. **Ablation Study and Discussions.** To study the effects of each component of the method, IdentifyMix on Cifar-10N with different noisy ratios was used in the experiment. Test accuracy is reported for IdentifyMix and results are analyzed in Figure 4 as follows. Several components of Figure 4 perform additive operations with the orange line representing the AUM sample selection mechanism, and the red line illustrating the AUM sample selection mechanism in combination with SSL training. The red line indicates the addition of contrastive learning. As can be seen on the green line, the overall framework of the algorithm IdentifyMix is incorporated, which also incorporates Mixup data augmentation.

Effect of semi-supervised learning. IdentifyMix relies heavily on semi-supervised learning as part of its algorithm. As can be seen from the comparison of AUM with AUM + SSL, SSL is capable of substantially improving the test accuracy of the Cifar-10N with different symmetric noisy ratios. It has been demonstrated that the former and the latter produce significant performance gains with an increase in noise rate, especially when the noise rate is 80%, the latter can produce substantial gain, demonstrating the effectiveness of semi-supervised learning.

TABLE 3. Comparison with state-of-the-art methods in test accuracy(%) on Animal-10N. Note that the best results are marked in **bold**.

Dataset	Animal-10N
Methods/Noise ratio	8%
Cross Entropy	79.40
AUM [19]	83.29
INCV [25]	79.54
BMM [26]	72.15
NLNL [23]	78.29
NCE+RCE [13]	71.48
NFL+RCE [13]	70.22
JoCoR [16]	82.82
MentorMix [6]	76.85
Co-learning [5]	84.77
ELR [27]	84.76
MOIT [28]	84.83
JNPL [24]	83.81
CLIM [33]	85.02
IdentifyMix(Ours)	**85.76**

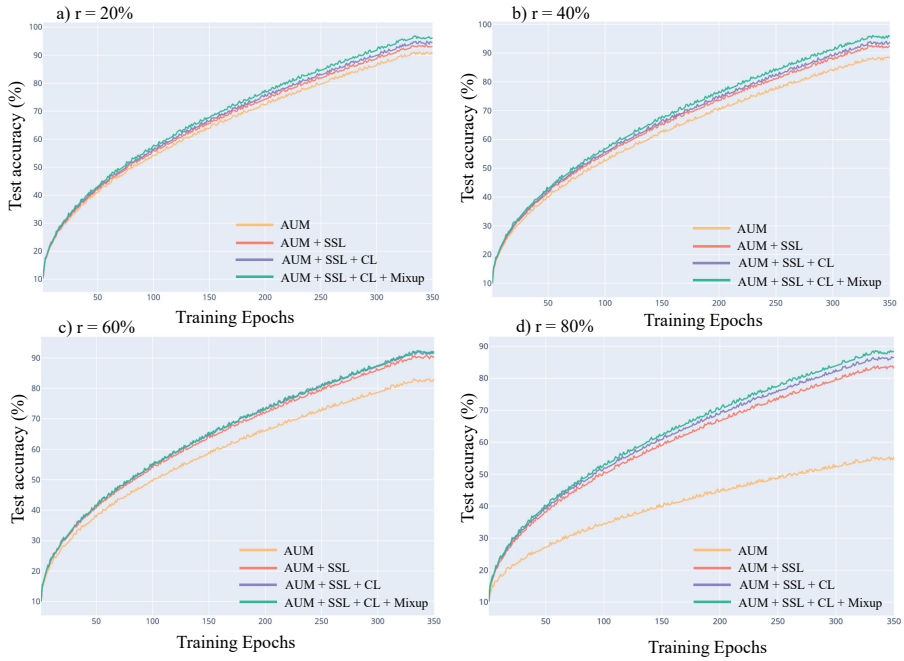

FIGURE 4. Test accuracy(%) on Cifar-10N with varying levels of symmetric noise for the ablation study.

Effect of Contrastive representation learning. Learning contrastive features is a crucial component of IdentifyMix. Figure 4 illustrates how contrastive learning impacts the performance of our method. Despite high levels of noise, it promotes performance enhancement by resisting the memory of noisy labels. The red line exhibits an approximate 3% drop in test accuracy respectively for Cifar-10N with an 80% noise rate.

Effect of Mixup data augmentation. It is impossible to overestimate the impact of the Mixup component on semi-supervised learning. In the green line, the test accuracy of Cifar-10N with 80% noise rate drops by approximately 2%. A possible explanation could be that Mixup is less effective due to class imbalances and a lack of labeled clean samples.

5. Conclusion

This paper proposed the IdentifyMix, a novel method to handle noisy labels in training data by a two-stage learning. AUM was introduced in the sample selection process to ensure clean labeled samples were accurately selected. Semi-supervised learning makes full use of large numbers of noise-labeled samples for effective noise-robust learning. Through contrastive learning and Mixup, the former compares data features in relation to limited samples with accurate labels. The latter provides data diversity relative to samples with clean labels. Both are concerned with reducing model memory noise and optimizing the performance of the model. The effectiveness of IdentifyMix was demonstrated by experiments conducted on multiple noisy datasets. The objective of future work is to extend our method to other tasks such as object detection and text matching.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61972097 and U21A20472, in part by the National Key Research and Development Plan of China under Grant 2021YFB3600503, in part by the Natural Science Foundation of Fujian Province under Grant 2021J01612 and 2020J01494, in part by the Major Science and Technology Project of Fujian Province under Grant 2021HZ022007, in part by the Industry-Academy Cooperation Project of Fujian Province under Grant 2018H6010, in part by the Fujian Collaborative Innovation Center for Big Data Application in Governments, and in part by the Fujian Engineering Research Center of Big Data Analysis and Processing. The authors also gratefully acknowledge the helpful comments and suggestions of the referees.

References

[1] K.J. Joseph, S. Khan, F.S. Khan, V.N. Balasubramanian, Towards open world object detection, In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5830-5840, 2021.

[2] H. Noh, S. Hong, B. Han, Learning deconvolution network for semantic segmentation, In: Proceedings of the IEEE International Conference on Computer Vision (CVPR), pp. 1520-1528, 2015.

[3] C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals, Understanding deep learning requires rethinking generalization, Communications of the ACM, 64 (2021), 107-115.

[4] X. Zhou, X. Liu, J. Jiang, X. Gao, X. Ji, Asymmetric loss functions for learning with noisy labels, In: International Conference on Machine Learning (ICML), pp. 12846-12856, 2021.

[5] C. Tan, J. Xia, L. Wu, S.Z. Li, Co-learning: Learning from noisy labels with self-supervision, In: Proceedings of the 29th ACM International Conference on Multimedia (MM), pp. 1405-1413, 2021.

[6] L. Jiang, D. Huang, M. Liu, W. Yang, Beyond synthetic noise: Deep learning on controlled noisy labels, In: International Conference on Machine Learning (ICML), pp. 4804-4815, 202.
[7] Y. Qu, S. Mo, J. Niu, Dat: Training deep networks robust to label-noise by matching the feature distributions, In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6821-6829, 2021.
[8] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I.W. Tsang, M. Sugiyama, Co-teaching: Robust training of deep neural networks with extremely noisy labels, Advances in Neural Information Processing Systems (NIPS), pp. 8536-8546, 2018.
[9] Z. Zhang, H. Zhang, S. O. Arik, H. Lee, T. Pfister, Distilling effective supervision from severe label noise, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9294-9303, 2020.
[10] Y. Liu, H. Guo, Peer loss functions: Learning from noisy labels without knowing noise rates, In; International Conference on Machine Learning (ICML), pp. 6226-6236, 2020.
[11] Y. Yao, T. Liu, B. Han, M. Gong, J. Deng, G. Niu, M. Sugiyama, Dual t: Reducing estimation error for transition matrix in label-noise learning, Advances in Neural Information Processing Systems, 33 (2020), 7260-7271.
[12] G. Patrini, A. Rozza, A.K. Menon, R. Nock, L. 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 (CVPR), 1944-1952, 2017.
[13] X. Ma, H. Huang, Y. Wang, S. Romano, S. Erfani, J. Bailey, Normalized loss functions for deep learning with noisy labels, International Conference on Machine Learning (ICML), pp. 6543-6553, 2020.
[14] Z. Zhang, M.R. Sabuncu, Generalized cross entropy loss for training deep neural networks with noisy labels, In: Advances in Neural Information Processing Systems (NIPS), pp. 8792-8802, 2018.
[15] J. Shu, Q. Xie, L. Yi, Q. Zhao, S. Zhou, Z. Xu, D. Meng, Meta-Weight-Net: Learning an explicit mapping for sample weighting, In: Advances in Neural Information Processing Systems (NIPS), pp. 1917-1928, 2019.
[16] H. Wei, L. Feng, X. Chen, B. An, 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 (CVPR), pp. 13726-13735, 2020.
[17] X. Yu, B. Han, J. Yao, G. Niu, I.W. Tsang, M. Sugiyama, How does disagreement help generalization against label corruption, International Conference on Machine Learning (ICML), pp. 7164-7173, 2019.
[18] H. Song, M. Kim, J.G. Lee, Selfie: Refurbishing unclean samples for robust deep learning, In: International Conference on Machine Learning (ICML), pp. 5907-5915, 2019.
[19] G. Pleiss, T. Zhang, E.R. Elenberg, K.Q. Weinberger, Identifying mislabeled data using the area under the margin ranking, Advances in Neural Information Processing Systems (NIPS), 33 (2020), 17044-17056.
[20] H. Lin, S. Wang, Z. Liu, S. Xiao, S. Du, W. Guo, FMixAugment for semi-supervised learning with consistency regularization, In: Chinese Conference on Pattern Recognition and Computer Vision (PRCV), pp. 127-139, 2021.
[21] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, C. Raffel, Mixmatch: A holistic approach to semi-supervised learning, In: Advances in Neural Information Processing Systems (NIPS), 5050-5060, 2019.
[22] H. Zhang, M. Cissé, Y.N. Dauphin, D. Lopez-Paz, Mixup: Beyond empirical risk minimization, In: International Conference on Learning Representations (ICLR), 2018.
[23] Y. Kim, J. Yim, J. Yun, J. Kim, NLNL: Negative learning for noisy labels, In: Proceedings of the IEEE/CVF International Conference on Computer Vision (CVPR), pp. 101-110, 2019.
[24] Y. Kim, J. Yun, H. Shon, J. Kim, Joint negative and positive learning for noisy labels, In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9442-9451, 2021.
[25] P. Chen, B. Liao, G. Chen, S. Zhang, Understanding and utilizing deep neural networks trained with noisy labels, International Conference on Machine Learning (ICML), pp. 1062-1070, 2019.
[26] E. Arazo, D. Ortego, P. Albert, N. E. O'Connor, K. McGuinness, Unsupervised label noise modeling and loss correction, In: International conference on machine learning (ICML), pp. 312-321, 2019.
[27] S. Liu, J. Niles-Weed, N. Razavian, C. Fernandez-Granda, Early-learning regularization prevents memorization of noisy labels, In: Advances in Neural Information Processing Systems, (NIPS), 33 (2020), 20331-20342.

[28] D. Ortego, E. Arazo, P. Albert, N.E. O'Connor, K. McGuinness, Multi-objective interpolation training for robustness to label noise, In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6606-6615, 2021.

[29] N. Karim, M.N. Rizve, N. Rahnavard, A. Mian, M. Shah, UNICON: Combating label noise through uniform selection and contrastive learning, In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9676-9686, 2022.

[30] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A simple framework for contrastive learning of visual representations, In: Proceedings of the 37th International Conference on Machine Learning (ICML), 1597-1607, 2020.

[31] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A.Maschinot, C. Liu, D. Krishnan, Supervised contrastive learning, In: Advances in Neural Information Processing Systems (NIPS), 18661-18673, 2020.

[32] Y. Tian, C, Sun, B. Poole, D. Krishnan, C. Schmid, P. Isola, What makes for good views for contrastive learning? In: Advances in Neural Information Processing Systems (NIPS), pp. 6827-6839, 2020.

[33] A. Ghosh, A.S. Lan, Contrastive learning improves model robustness under label noise, In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2703-2708, 2021.

[34] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. Raffel, E.D. Cubuk, A. Kurakin, C.L. Li, FixMatch: Simplifying semi-supervised kearning with consistency and confidence, In: Advances in Neural Information Processing Systems (NIPS), pp. 596-608, 2020.

[35] E.D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, Q.V. Le, Autoaugment: Learning augmentation policies from data, CoRR, 1805.09501, 2018.

[36] S. Wang, X. Lin, Z. Fang, S. Du, G. Xiao, Contrastive consensus graph learning for multi-view clustering, IEEE/CAA J. Automatica Sinica 9 (2022), 2027-2030.

[37] M. Ciortan, R. Dupuis, T. Peel, A framework using contrastive learning for classification with noisy labels, Data 6 (2021), 61.

[38] J. Han, P. Luo, W. Wang, Deep self-learning from noisy labels, In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 5138-5147, 2019.