# Late Stopping: Avoiding Confidently Learning from Mislabeled Examples

Suqin Yuan[1] Lei Feng[2*] Tongliang Liu[1*]
[1]The University of Sydney [2]Nanyang Technological University

## Abstract

*Sample selection is a prevalent method in learning with noisy labels, where small-loss data are typically considered as correctly labeled data. However, this method may not effectively identify clean hard examples with large losses, which are critical for achieving the model's close-to-optimal generalization performance. In this paper, we propose a new framework, Late Stopping, which leverages the intrinsic robust learning ability of DNNs through a prolonged training process. Specifically, Late Stopping gradually shrinks the noisy dataset by removing high-probability mislabeled examples while retaining the majority of clean hard examples in the training set throughout the learning process. We empirically observe that mislabeled and clean examples exhibit differences in the number of epochs required for them to be consistently and correctly classified, and thus high-probability mislabeled examples can be removed. Experimental results on benchmark-simulated and real-world noisy datasets demonstrate that the proposed method outperforms state-of-the-art counterparts.*

## 1. Introduction

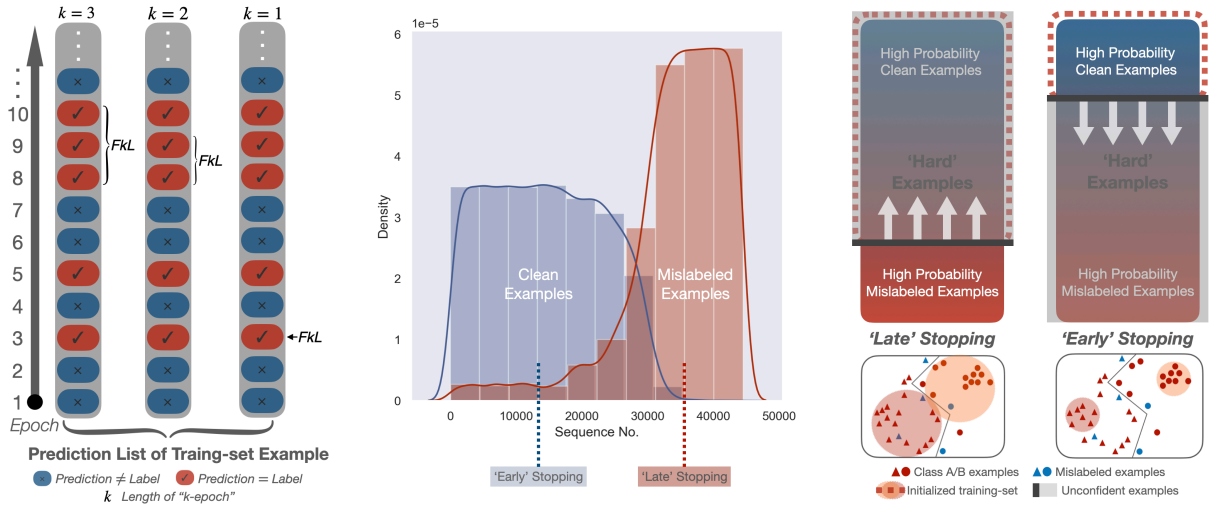
Deep Neural Networks (DNNs) have achieved outstanding success in various tasks, while the success largely relies on data with high-quality annotations [13, 55, 41, 19]. In many real-world scenarios, it would be quite difficult to collect large-scale accurately labeled data, which could inevitably contain noisy labels. Unfortunately, previous studies [1, 56] showed that DNNs can easily overfit random labels, resulting in poor generalization performance. Therefore, an increasing number of methods have been proposed for Learning with Noisy Labels (LNL).

One mainstream solution in existing methods of LNL is to train the classifier with confident examples [20, 14, 46, 34, 36], which is based on the *memorization effect* of DNNs, i.e., DNNs learn example with dominant patterns first and then overfit rare ones [1]. Given only noisy data, to exploit the memorization effect, the typical strategy starts from a small confident clean dataset and then gradually expands the dataset, which prevents DNNs from over-fitting noisy data. In general, there are two primary approaches to exploit confident examples in the learning process. The first approach involves identifying examples with high-probability clean labels and training the classifier based on these examples, which is commonly referred to as the "small-loss trick" [20, 14, 12, 28, 52]. The second approach involves controlling the learning process of the classifier to primarily learn high-probability clean examples in noisy datasets, which is commonly referred to as "early stopping" [37, 42, 17, 25, 3].

Despite providing satisfactory performance, these approaches present an unintended consequence in their methods to prevent over-fitting noise. Specifically, to reduce the impact of mislabeled examples, these approaches limit the capability of the model to effectively learn *clean hard examples* (CHEs) in noisy datasets, where CHEs are defined as clean examples that are close to the decision boundary, and a significant proportion of CHEs are non-dominated sub-population examples [9]. Identifying CHEs in noisy data is quite challenging [2, 22], as both the CHEs and the mislabeled examples are often characterized by large losses [52, 11, 6], causing them to become entangled. To maintain the purity of the dataset of confident examples, existing LNL methods [34, 2] normally try to eliminate potential examples that are likely to be mislabeled, which inevitably contain many CHEs. However, it is necessary to consider the positive impact of memorization effects on underrepresented sub-populations [10] (i.e., CHEs) when learning from natural datasets, as this is crucial for achieving close-to-optimal generalization performance.

In this work, our goal is to enable the classifier to learn as many useful non-dominated sub-population examples in the training set as possible during the process of learning with noisy labels. This entails selecting as many clean examples as possible, particularly the clean hard examples, during the sample selection process. In relation to this context, we introduce a novel concept termed *First-time k-epoch Learning* (FkL), which is defined as the index of the epoch during the training procedure where an example has been predicted to its given label for consecutive $k$ epochs for the first time, as shown in Figure 1(a).

*Corresponding authors.




(a) (b) (c)

Figure 1. (a) We propose the *First-time k-epoch Learning* (FkL) metric, which determines the minimum index of the training epoch until which an example has been predicted as its given label for consecutive $k$ epochs. (b) The normalised histogram of CIFAR-10 examples with 40% symmetric label noise w.r.t. the sequence they meet the FkL metric during training procedure. The horizontal axis represents the sequential order in which training examples meet the FkL metric. The vertical axis represents the normalised histogram of examples. (c) Rather than the methods that start from a small yet clean training set ('Early' Stopping), our proposed framework starts from a large training set ('Late' Stopping) that retains as many clean examples as possible.

Using *First-time k-epoch Learning* (FkL) as a metric, we sequence the examples in the noisy dataset according to the order they meet the FkL metric during the training procedure, as shown in Figure 1(b). As shown in the same figure, most mislabeled examples can only be classified into their given labels for consecutive $k$ epochs in the later stages of training (i.e., larger in the sequence), which implies a relatively large FkL value. This observation suggests that the examples with large FkL values (i.e., those examples that are classified to their given labels for consecutive $k$ epochs only in the late training stage) are predominantly those with incorrect labels. Therefore, the FkL values of different training examples can be used to distinguish whether an example is mislabeled or not.

Motivated by the above observations, we propose a novel method based on reverse thinking of the conventional confident example selection strategy. Our proposed method, called *Late Stopping*, employs an iterative sample selection process that gradually reduces the noise rate of the dataset, leading to a positive feedback loop. Instead of selecting high-probability clean examples in the early stage, our method focuses on *selecting high-probability mislabeled examples in the late stage* to retain as many CHEs as possible in the training set throughout the learning process, even though this method may lead to the retention of an acceptable level of mislabeled examples, as shown in 1(c). Building on this, we introduce a novel sample selection criterion, FkL, which works effectively in selecting mislabeled examples under the *Late Stopping* framework, and empirical results show that the FkL is more effective than the *loss* criterion. We evaluate our method on benchmark-simulated and real-world noisy datasets. Empirical results demonstrate that the *Late Stopping* method achieves superior performance compared with state-of-the-art counterparts of learning with noisy labels.

## 2. Related Work

Existing methods of learning with noisy labels can be broadly categorized into two types. The first category includes classifier-consistent algorithms that use the noise transition matrix [27, 29, 54, 35, 8], which indicates the probabilities of clean labels flipping to noisy labels. The second category focuses on heuristic methods to mitigate the impact of label noise [36, 50, 39, 45, 30, 49, 36], which is also the primary focus of our work. Many existing heuristic methods in LNL are based on the memorization effect, where the algorithm attempts to only learn from confident clean examples [20, 14, 46, 34]. Alternatively optimizing the classifier and updating the training set is not a new idea in LNL. For instance, Joint Optim [42], Co-teaching [14], SELF [34], and Me-Momentum [2] employ a similar positive feedback loop to our proposed *Late Stopping*.

Previous research on the dynamics of DNNs has demonstrated that hard examples are typically learned during the late stage of the learning process [1, 43, 31]. Furthermore, many recent studies have employed various training-time metrics to quantify the "hardness" of examples [32, 16, 7, 4, 21, 31], leading to an increase in LNL approaches that use learning dynamics to select clean examples. For instance,

**Algorithm 1** Late Stopping

**Input:** Original noisy training set $\mathcal{D}_1$, iteration rate $m\%$, noise rate $n\%$, epoch $T_{max}$ and iteration $I_{max}$.
**Output:** Extracted final training set and the classifier.

1: **for** $I$ = 1, ..., $I_{max}$ **do**
2: **Initialize** $\mathcal{D}_i$, $S_{F_i}$ = 0 and $\mathcal{D}_{F_i}$ = [ ].
3: //Initialize new training set, the number/dataset of examples that meet the *First-time k-epoch Learning*, respectively.
4: **for** $T$ = 1, ..., $T_{max}$ **do**
5: **Train** $f_i$ on $\mathcal{D}_i$.
6: **Update** $\mathcal{D}_{F_i}$, $S_{F_i}$.
7: **Break** and **Output** $\mathcal{D}_{i+1} = \mathcal{D}_{F_i}$, if $S_{F_i} > S_{F_{i-1}} \times (1 - m\%)$.
8: **end for**
9: **Break** and **Output** $f_i$ and $\mathcal{D}_{F_i}$, if $m \times i > n$.
10: **end for**

FSLT&SSFT [31], Self-Filtering [48], SELFIE [40], and RoCL [57] adopt a criterion similar to our proposed FkL criterion. Specifically, FSLT&SSFT and Self-Filtering use the classifier's error prediction to pinpoint clean examples, SELFIE employs entropy values from prediction histories for selection, and RoCL leverages the variance of training losses to select clean examples. Instead of quantifying the "hardness" of examples based on the characteristics of individual examples in the dataset, our proposed method focuses on extracting the intrinsic robust learning ability of DNNs from the model trained on noisy datasets.

## 3. Late Stopping

In this section, we elaborate on our proposed *Late Stopping* method. As shown in Figure 1, the primary goal of the *Late Stopping* framework is to maximize the retention of clean examples in the training set throughout the learning process, enabling our method to learn clean hard examples (CHEs) from a noisy training set. Therefore, we halt the training process in the late training stage, leveraging *First-time k-epoch Learning* (FkL) to exploit the results of the intrinsic robust learning ability of DNNs for sample selection. Iteratively performing this operation allows for a gradual reduction of noise while maximizing the retention of clean samples in the training dataset.

### 3.1. Algorithm Flow

For a comprehensive explanation of *Late Stopping*, we begin by formally defining *First-time k-epoch Learning* (FkL). In learning with noisy labels, let us consider a model $f_i$ trained on a noisy training set $\mathcal{D}_i$ composed of $n$ training examples $\{\mathbf{x}_j, y_j\}_{j=1}^{n}$ where $y_j$ denotes the given label (which may not be true) of $\mathbf{x}_j$. After $t$ training epochs, the predicted label $\hat{y}_j^t$ for the instance $\mathbf{x}_j$ can be obtained. Let $\text{acc}_j^t = \mathbb{1}_{\hat{y}_j^t = y_j}$ denote a binary variable indicating whether $f_i$ predicts the given label $y_j$ of the instance $\mathbf{x}_j$ at epoch $t$. The FkL for the instance $\mathbf{x}_j$ is defined as the the minimum index of the training epoch that the instance $\mathbf{x}_j$ has been predicted to its given label $y_j$ for $k$ consecutive epochs:

$$\text{FkL}_j = \underset{t^* \in t}{\text{argmin}} \left( \text{acc}_j^{t^*} \wedge ... \wedge \text{acc}_j^{(t^*-k+1)} = 1 \right).$$

If $\text{FkL}_j = t^*$, it implies that the classifier $f_i$ "learns" the instance $\mathbf{x}_j$ after $t^*$ epochs. Notably, if $\mathbf{x}_j$ is "learned" by the classifier early in the training, it will have a small FkL value, however, if it is "learned" in the late stage, it will have a large FkL value. Based on our FkL definition, we observed that the majority of examples with large FkL values are mislabeled, as shown in Figure 1(a). This observation provides a rationale for selecting examples with large FkL values, those learned in the late training stage, as high-probability mislabeled examples.

With the FkL selection criterion, we present the algorithmic framework of *Late Stopping*. At the low level (Steps 4-8 of Algorithm 1), the sample selection strategy follows a "from easy to hard" curriculum learning procedure [5, 18]. During the $i$-th iteration, the classifier $f_i$ is trained on a new training set $\mathcal{D}_i$ (Step 5), and newly identified FkL examples are added to $\mathcal{D}_{F_i}$ (Step 6). If the size of $\mathcal{D}_{F_i}$, represented as $S_{F_i}$, exceeds a predefined threshold, the training in this iteration will halt, and the output becomes $\mathcal{D}_{F_i}$ (Step 7). At the higher level (Steps 1-10 of Algorithm 1), our method learns through a "from hard to easy" positive feedback loop, distinguishing it from most existing LNL approaches. The algorithm starts with initializing $\mathcal{D}_1$ and training $f_1$ (Iteration 1). In subsequent iterations, the new training set $\mathcal{D}_i$ (Step 2, Iteration $i$) is updated from the previous iteration's $\mathcal{D}_{F_{i-1}}$ (Step 8, Iteration $i-1$). When the stopping condition is met, the output is $f_i$ and $\mathcal{D}_{F_i}$ (Step 10).

### 3.2. Learning CHEs by Positive Feedback Loop

Here, we delve deeper into the typical positive feedback loop in LNL, highlighting its limitations in effectively learning CHEs. We then elucidate how our proposed method overcomes these limitations and maximizes the retention of clean examples throughout the learning process.

**Typical positive feedback loop in LNL.** According to statistical learning theory [33], training data with better quality yields a better classifier. Therefore, we can improve the classifier's generalization performance in LNL through the construction of a positive feedback loop. The goal of the positive feedback loop is that, after each training step, the new classifier showcases enhanced generalization performance compared with its predecessor. By doing so, it can more robustly guide the sample selection process, enabling the use of improved training data for classifier training in the succeeding steps [2].

Most LNL methods, relying on confident examples, initiate with either a small yet clean training set or a weak yet reliable classifier. To enhance the classifier's generalization performance during initialization, they often rigorously limit memorization effects and eliminate potential mislabeled examples, even if this entails removing many CHEs. By restricting the memorization of mislabeled examples, they achieve superior generalization performance compared with unrestricted training on the original set. Then, they either iteratively or incrementally, train classifiers, ultimately obtaining either a larger clean training set or a more robust classifier. This strategy is effective when the classes or subclasses within the dataset are well-balanced and the examples within each class are interrelated, allowing the classifier to learn harder examples from easier ones.

**Limitations of typical positive feedback loop.** Although LNL methods that rely on a typical positive feedback loop achieve good generalization performance in a variety of benchmark tasks, natural datasets are often highly imbalanced. This imbalance poses a challenge to achieving close-to-optimal generalization performance with existing LNL methods. Specifically, such challenges are because CHEs are often characterized by large losses and non-dominant patterns. As a result, substantial learning about CHEs predominantly transpires during the late stages of training, driven by memorization effects [2, 52]. The conventional positive feedback loop, aimed at preventing overfitting, often strictly limits the memorization effects of DNNs. Consequently, this strict constraint reduces a classifier's ability to learn from under-represented subpopulations in datasets [9], and thus limits its learning primarily to typical (dominant) classes and subclasses. Such constraints ultimately lead to a situation where the learning ability of rare classes and subclasses, i.e., CHEs, is continuously neglected, while the learning ability of the classes and subclasses representing the dominant pattern is continuously enhanced.

**Positive feedback loop in *Late Stopping*.** Based on our previous discussions, to achieve close-to-optimal generalization performance on natural datasets, we need to focus on learning from CHEs throughout the training process. To achieve this, we prolong the training process for each training iteration to make use of the memorization effects to help memorize CHEs. With a loose sample selection process, we can ensure that we maximize the retention of clean examples in the training set throughout the learning process. However, retaining as many clean examples as possible during the training initialization requires accepting a significant amount of label noise in the initial training set. Additionally, relying on the "memorization effects in the late training stage" to learn CHEs might result in poor generalization performance, as the classifier may seriously overfit label noise in this stage.

To counter these negative effects, we propose a "from hard to easy" positive feedback loop within the Late Stopping framework. While we prolong the training process in each training iteration, we simultaneously use the classifier's robust learning capabilities as the criterion for sample selection, i.e., FkL. In each iteration, we train a new classifier on a new training set, using it solely to guide the sample selection for the current iteration. In this positive feedback loop, we opt to disregard the generalization performance of the classifier in each iteration. Instead, we exploit the robust learning capabilities of each new classifier, $f_i$, in its $i$ iteration on $\mathcal{D}_i$ from its learning dynamics, i.e., using the FkL criterion to select examples for the next iteration.

The new training set, obtained by the FkL criterion from the previous iteration, will have less noise. This enables the training of a better new classifier, which can learn more from the CHEs before it begins to memorize mislabeled examples. Therefore, tracking learning dynamics using the FkL criterion can better distinguish between CHEs and mislabeled examples, resulting in a new training set with even less noise than the previous one but contains CHEs. By observing each classifier's learning dynamics through FkL in every iteration, we can iteratively refine our training set, reducing noise while retaining the majority of clean examples, as shown in Figure 2.

**Discussion.** Our proposed method is motivated by the seemingly conflicting views on the ability of DNNs to handle label noise as presented in recent studies: "Deep neural networks easily fit random labels [56]" and "Deep learning is robust to massive label noises [37]". Empirical evidence [1] suggests that DNNs first learn simple patterns before over-fitting in the late training stage. However, there is a scarcity of studies on the behavior of DNNs during the intermediate stages of training, which may be a critical factor for understanding the robustness of DNNs to label noise - DNNs might exhibit spontaneously robust learning to massive label noises in all periods of training, though the generalization performance is over-fitting noisy labels. Thus, we aim to separate the robust learning ability of DNNs from their generalization performance. To this end, we leverage the intrinsic robust learning ability of DNNs by continuing the training process until the generalization performance se-

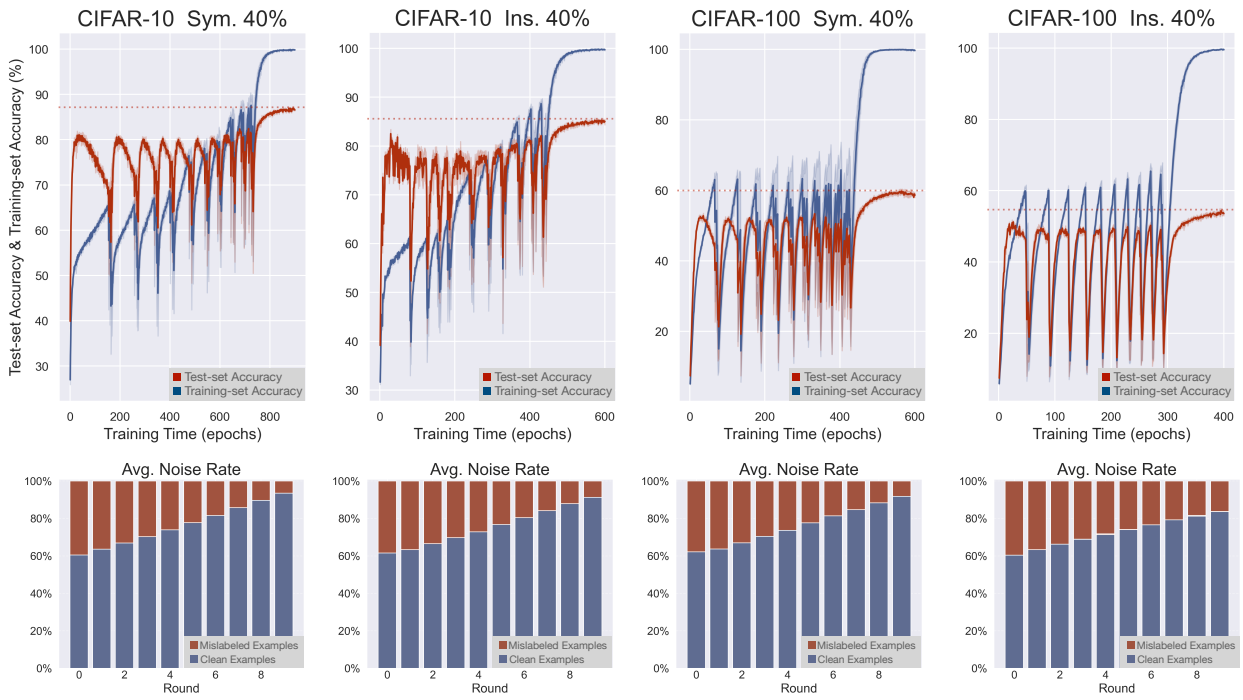



Figure 2. We refer to one update of the classifier and the training set as one iteration. Top subfigures: We illustrate the changes in the Test-set Accuracy and the Training-set Accuracy of the classifier during the training process of *Late Stopping*. There are ten peaks in each figure because we set the number of iterations = 10 and the classifiers are re-initialized at the beginning of each new iteration. Bottom subfigures: We illustrate the changes in the noise rate of the training set in each iteration. There are ten bars in each figure because we set the number of iterations = 10 and the training set of each iteration is obtained from the previous iteration.

riously deteriorates (i.e., *Late Stopping*) and we extract the robust learning results during this process (i.e., FkL).

## 4. Experiments

In this section, we conduct experiments on both synthetic and real-world datasets containing label noise to validate the effectiveness of our proposed *Late Stopping*.

**Datasets.** We use two popular benchmark datasets i.e., CIFAR-10 and CIFAR-100 [23], to test the generalization performance of our proposed method. Following previous works [2, 51], we manually corrupt CIFAR-10 and CIFAR-100 with Symmetric noise (abbreviated as Sym.) [44] and Instance-dependent (abbreviated as Ins.) noise [53]. For Sym. noise and Ins. noise, the noise rate is set to 20% and 40%. For a fair comparison, we leave out 10% of the noisy training data. We employ the real-world noisy dataset CIFAR-10N [47], which equips the training datasets of CIFAR-10 with human-annotated real-world noisy labels. More details about the above datasets and noise types are provided in Appendix A1.

**Compared methods.** We compare the *Late Stopping* method (Algorithm 1) with the following well-known methods: (1) Statistically inconsistent approaches (small-loss trick): MentorNet [20], Co-teaching [14], and JoCoR [46]; (2) Statistically inconsistent approaches (other tricks): Joint Optim [42], CDR [51], and Me-Momentum [2]; (3) Statistically consistent approaches: Forward [35] and DMI [54]. Note that drawing a direct comparison with certain semi-supervised approaches, especially those with multiple technical aggregations such as SELF [34], DivideMix [24], ELR+ [26], and Unicon [22]. The results for baselines are copied from original papers and [2, 47].

**Network structure and experimental setup.** We utilize a typical warming-up strategy to ensure stable predictions of the DNNs. The optimizer settings used in our experiments are as follows: SGD with a momentum of 0.9, weight decay of 5e-4, batch size of 128, and initial learning rate of 0.02. In our experiments, we applied typical data augmentations such as horizontal flipping and random cropping. For the experiments on synthetic noisy datasets, ResNet-18 and ResNet-34 [15] are used for CIFAR-10 and CIFAR-100, respectively. For the experiments on real-world noisy datasets, ResNet-34 is used for CIFAR-10N. More details about our experimental setup are provided in Appendix A2.

### 4.1. Effectiveness of Late Stopping

Here, we aim to provide empirical evidence to verify the effectiveness of *Late Stopping*. We conduct experiments on the CIFAR-10 and CIFAR-100 datasets, as their

Table 1. Label precision of the selected examples ($45k$ in total, with $27k$ clean examples).

| | | Selecting Clean Examples | | | Selecting Mislabeled Examples | | |
|---|---|---|---|---|---|---|---|
| Data&Noise | criterion | 0-10$k$ | 0-15$k$ | 0-25$k$ | 30$k$-45$k$ | 35$k$-45$k$ | 40$k$-45$k$ |
| CIFAR-10 *Sym. 40%* | *loss* | 92.73% | 89.56% | 82.89% | 75.67% | 83.49% | 91.54% |
| | FkL(Ours) | **95.49%** | **95.59%** | **93.56%** | **96.39%** | **99.81%** | **99.94%** |
| CIFAR-10 *Ins. 40%* | *loss* | 75.55% | 72.33% | 71.38% | 60.27% | 68.69% | 77.28% |
| | FkL(Ours) | **81.88%** | **81.67%** | **81.51%** | **78.51%** | **88.85%** | **99.30%** |
| CIFAR-100 *Sym. 40%* | *loss* | 88.43% | 84.72% | 78.49% | 68.02% | 75.73% | 84.68% |
| | FkL(Ours) | **97.45%** | **96.78%** | **95.06%** | **94.63%** | **97.91%** | **98.86%** |
| CIFAR-100 *Ins. 40%* | *loss* | 71.30% | 68.23% | 65.29% | 45.91% | 44.23% | 42.56% |
| | FkL(Ours) | **89.25%** | **88.56%** | **84.40%** | **78.22%** | **86.28%** | **91.76%** |

Table 2. Numbers of clean examples of the final training set ($27k$ in total).

| | CIFAR-10 (*Sym. 40%*) | CIFAR-10 (*Ins. 40%*) | CIFAR-100 (*Sym. 40%*) | CIFAR-100 (*Ins. 40%*) |
|---|---|---|---|---|
| Me-Momentum [2] | 25,791 | 25,694 | 22,244 | 20,779 |
| Late Stopping (Ours) | **26,562** | **25,915** | **26,120** | **23,979** |

ground-truth labels are available. The average results of five runs of our experiments are presented in Figure 2, which demonstrates the iterative improvement in the quality of the training set when *Late Stopping* is applied in the learning process. And a classifier with better generalization performance is obtained. We demonstrate the ability of our proposed FkL criterion to select mislabeled examples under the *Late Stopping* framework, compared with the *loss* criterion. Additionally, we show the effectiveness of our proposed *Late Stopping* in retaining clean examples in the training set by comparing it with Me-Momentum [2], which is also an sample selection method that focuses on extracting clean hard examples.

**Comparison with loss criterion.** The selection of confident clean examples based on the *small-loss* criterion is a commonly used approach in LNL and has been shown to be effective [20, 14, 46]. However, the approach of selecting mislabeled examples based on the *large-loss* [6] can be tricky since it fails to distinguish between mislabeled and clean hard examples [52]. We conducted experiments on the CIFAR-10 and CIFAR-100 datasets to compare the effectiveness of the FkL and *loss* criterion for selecting examples during the same training process. Specifically, we ranked all examples ($45k$ in total, with $27k$ clean examples) using the FkL criterion and the *loss* criterion, respectively, from the smallest $0k$ to the largest $45k$. We then compared the precision of the two criteria in selecting clean examples and mislabeled examples in different ranges. The details of the experiments and the results are presented in Table 1. Our experimental results demonstrate that the FkL criterion outperforms the *loss* criterion in selecting both clean and mislabeled examples under the *Late Stopping* framework. The largest performance gap was observed in the $40k$-$45k$ range. These results provide empirical evidence that the FkL criterion is more effective than the *loss* criterion for selecting mislabeled examples in the *Late Stopping* framework.

**Comparison with hard example selection approach.** In Table 2, we compare our proposed method with Me-Momentum [2] which also focuses on selecting clean hard examples. Since there are no clear definitions of clean hard examples, we cannot directly compare the effectiveness of selecting such examples. Nevertheless, we can make an indirect comparison by comparing the ability to select clean examples. Our experimental results demonstrate that *Late Stopping* is an effective approach for retaining clean hard examples. On the CIFAR-10 dataset, both *Late Stopping* and Me-Momentum showed comparable performance. However, on the more challenging CIFAR-100 dataset, *Late Stopping* significantly achieved better performance in selecting clean hard examples.

### 4.2. Classification Accuracy

**Synthetic datasets.** The experimental results on test accuracy on synthetic datasets with class-dependent and instance-dependent label noise are provided in Table 3 and 4. Each trial is repeated five times and the mean value and standard deviation are recorded. On CIFAR-10, our method achieves varying degrees of lead over baselines. In the 20% symmetric noise task, which is the simplest task in all cases, the Me-Momentum baseline outperforms ours, which illustrates the effectiveness of conventional sample selection based on confident clean examples for simple LNL. For the more challenging CIFAR-100 dataset, our proposed method consistently achieved the best results. The size of each class in CIFAR-100 is ten times smaller than that of CIFAR-10, making it difficult to retain sufficient CHEs while maintaining a small yet clean training set using conventional sample selection methods. Methods that focus on improving the

Table 3. Test performance (mean±std) of each approach using ResNet-18 on CIFAR-10.

| | Sym. 20% | Sym. 40% | Ins. 20% | Ins. 40% |
|---|---|---|---|---|
| Late Stopping(Ours) | 91.06±0.22% | **88.92±0.38%** | **91.08±0.23%** | **87.41±0.38%** |
| Cross-Entropy [38] | 85.00±0.43% | 79.59±1.31% | 85.92±1.09% | 79.91±1.41% |
| MentorNet [20] | 80.49±0.11% | 77.48±3.45% | 79.12±0.42% | 70.27±1.52% |
| Forward [35] | 85.63±0.11% | 74.30±0.26% | 85.29±0.38% | 74.72±3.24% |
| Co-teaching [14] | 87.16±0.52% | 83.59±0.28% | 86.54±0.11% | 80.98±0.39% |
| JoCoR [46] | 88.69±0.19% | 85.44±0.29% | 87.31±0.27% | 82.49±0.57% |
| DMI [54] | 88.18±0.13% | 83.98±0.48% | 89.14±0.36% | 84.78±1.97% |
| Joint Optim [42] | 89.70±0.36% | 87.79±0.20% | 89.69±0.42% | 82.62±0.57% |
| CDR [51] | 89.68±0.38% | 86.13±0.44% | 90.24±0.39% | 83.07±1.33% |
| Me-Momentum [2] | **91.44±0.33%** | 88.39±0.34% | 90.86±0.21% | 86.66±0.91% |

Table 4. Test performance (mean±std) of each approach using ResNet-34 on CIFAR-100.

| | Sym. 20% | Sym. 40% | Ins. 20% | Ins. 40% |
|---|---|---|---|---|
| Late Stopping(Ours) | **68.67±0.67%** | **64.10±0.40%** | **68.59±0.70%** | **59.28±0.46%** |
| Cross-Entropy [38] | 57.59±2.55% | 45.74±2.61% | 59.85±1.56% | 43.74±1.54% |
| MentorNet [20] | 52.11±0.10% | 35.12±1.13% | 51.73±0.17% | 40.90±0.45% |
| Forward [35] | 57.75±0.37% | 38.59±1.62% | 58.76±0.66% | 44.50±0.72% |
| Co-teaching [14] | 59.28±0.47% | 51.60±0.49% | 57.24±0.69% | 45.69±0.99% |
| JoCoR [46] | 64.17±0.19% | 55.97±0.46% | 61.98±0.39% | 50.59±0.71% |
| DMI [54] | 58.73±0.70% | 49.81±1.22% | 58.05±0.20% | 47.36±0.68% |
| Joint Optim [42] | 64.55±0.38% | 57.97±0.67% | 65.15±0.31% | 55.57±0.41% |
| CDR [51] | 66.52±0.24% | 60.18±0.22% | 67.06±0.50% | 56.86±0.62% |
| Me-Momentum [2] | 68.03±0.53% | 63.48±0.72% | 68.11±0.57% | 58.38±1.28% |

Table 5. Test accuracy of each approach using ResNet-34 on CIFAR-10N (*Worst*).

| Cross-Entropy [38] | Forward [35] | Co-teaching [14] | JoCoR [46] | Me-Momentum [2] | Late Stopping(Ours) |
|---|---|---|---|---|---|
| 77.69±1.55% | 79.79±0.46% | 83.83±0.13% | 83.37±0.30% | 84.71±0.37% | **85.24±0.38%** |

number of CHEs in the training set can achieve better generalization performance in this task. For instance, DMI and Co-teaching show a much larger performance gap with our proposed method on CIFAR-100 compared with CIFAR-10. Furthermore, Table 2 validates our method's superior capability to retain clean examples, resulting in a larger performance gap with Me-Momentum on CIFAR-100.

**Real-world dataset.** To validate the efficacy of our proposed method on real-world datasets, we assess our method using the CIFAR-10N dataset in Table 4, a recognized benchmark in learning with noisy labels tasks with both ground-truth labels and human-annotated real-world noisy labels. We utilized the most challenging real-world noisy labels from CIFAR-10N, i.e., the "*Worst*" setting. In this setting, for each image, if there are any incorrectly labeled examples, the given label is randomly selected from human-annotated false labels. Each trial is repeated five times and the mean value and standard deviation are recorded. As shown in Table 4, our method achieves the best performance compared with other methods.

### 4.3. Further analysis

Here, we conduct further analysis of our proposed method with the same experimental settings as those used in Section 4.2 unless stated otherwise.

**Limitations.** While our method achieves good robustness by simply removing the lately learned examples, it is not immune to errors. Some clean examples might be incorrectly removed from the training set, i.e., *falsely removed examples*, and some mislabeled examples are incorrectly retained in the training set, i.e., *falsely retained examples*. We visualize such typical examples in Figure 3 and provide the following empirical analysis.

*Falsely retained examples* are mislabeled with the wrong label $L_w$, however, such examples often possess patterns similar to the dominant subclasses in the $L_w$ class, making it difficult to identify them as high probability mislabeled examples by the FkL criterion. As shown in Table 6, based on the FkL criterion and the loss criterion, the "hardness" of these examples decreases after being mislabeled. In particular, we named the examples with a decrease in

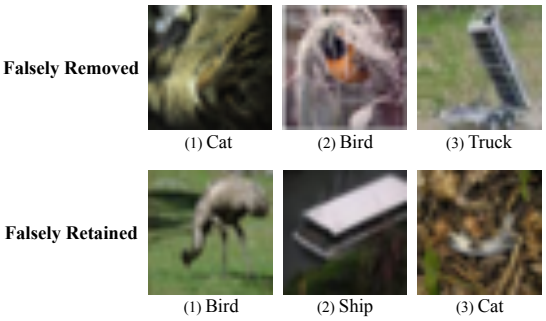



Figure 3. The labels in the figure are ground-truth labels for each example. Falsely Removed (1-3) are typical examples that are incorrectly removed in the first iteration of Late Stopping; Falsely Retained (1-3) are typical examples that are incorrectly marked by FkL in the last iteration of Late Stopping. In the training set, Falsely Retained (1) is incorrectly labeled as "deer", Falsely Retained (2) is incorrectly labeled as "truck", and Falsely Retained (3) is incorrectly labeled as "frog".

Table 6. The comparison of the average ranking of *falsely retained examples* using the *loss* criterion and the FkL criterion before and after fixing the noisy labels (CIFAR-10, Before: *Sym. 40%* noise).

| Label | Criterion | *Avg.* Ranking |
|---|---|---|
| Before fixing (Given label) | FkL | 22340.66 |
| | *loss* | 23673.50 |
| After fixing (Ground-truth label) | FkL | 28452.08 (**+27.36%**) |
| | *loss* | 29476.46 (**+24.51%**) |

Table 7. Comparison of the number of clean examples in datasets before and after applying our method as a pre-processing approach to decrease the noise level of a 40% noisy training set to 20%.

| Training set | Before (*40% noise*) | After (*20% noise*) |
|---|---|---|
| CIFAR-10 (*Sym.*) | 27228 | 27139 (**-0.33%**) |
| CIFAR-10 (*Ins.*) | 27094 | 26710 (**-1.42%**) |
| CIFAR-100 (*Sym.*) | 27341 | 27037 (**-1.11%**) |
| CIFAR-100 (*Ins.*) | 27249 | 25079 (**-7.96%**) |
| CIFAR-10N (*Worst*) | 29896 | 29613 (**-0.95%**) |

training losses after being mislabeled as "*hard mislabeled examples*". Our experimental results indicate that *hard mislabeled examples* pose a challenge for all sample selection-based LNL methods. More details about Table 6 are provided in Appendix B1.

*Falsely removed examples* refer to the examples that are too hard to remove in practice. To limit the computational resources consumed by our proposed method, we set the removal rate (Algorithm 1, iteration rate $m\%$) to a large value. Thus, our proposed method cannot completely distinguish between rare patterns and mislabeled examples, which can result in a small number of examples from rare subclasses being mistakenly removed from the training set.

Table 8. Comparison of the total training hours on CIFAR-100.

| Co-teaching [14] | Me-Momentum [2] | Late Stopping (Ours) |
|---|---|---|
| 3.3h | 7.1h | 9.5h |

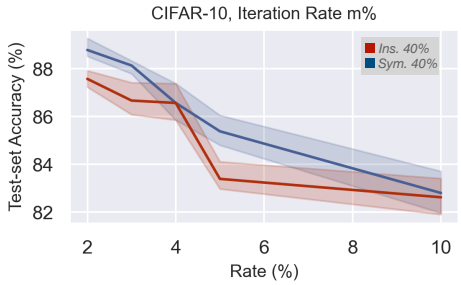



Figure 4. Illustrates the performance of classifiers with the change in iteration rate $m$.

**Versatility.** We are focusing on demonstrating the concept, i.e., using *Late Stopping* to retain clean hard examples in the training set, but not on boosting the classification performance. However, to cope with even more complex noisy scenarios, our method works well in combination with other advanced techniques. In particular, our approach has excellent performance as a pre-processing approach for reducing the noise rate of the noisy training set. As shown in Table 7, we evaluate *Late Stopping* as a pre-processing approach to decrease the noise level of a 40% noisy training set to 20%. Our experimental results indicate that our proposed method performs exceptionally well in reducing label noise. It can substantially reduce the noise level of the training set with minimal loss of clean examples.

**Training time.** Our proposed method involves training the classifier on a larger training set for a longer period, which may not be advantageous in terms of efficiency. Despite this, as the noise rate decreases, the size of the training set decreases, and the speed of obtaining FkL-examples is accelerated (see Figure 2). As shown in Table 8, we compare the training time with typical baseline methods on CIFAR-100 (*Sym. 40%*). Our experimental results indicate that the training time of our proposed method is in a comparable range with other approaches.

**Sensitivity of the iteration rate.** The iteration rate $m$ (see Algorithm 1) determines the iteration number for *Late Stopping* and the number of examples removed in each iteration. To investigate the sensitivity of $m$, we conducted experiments on CIFAR10 with 40% Sym. noise and Ins. noise by varying $m$ in the range $\{10, 5, 4, 3, 2\}$. Figure 4 shows the performance of classifiers, which gradually improves as $m$ decreases. This observation aligns with the findings presented in Table 1, where a decrease in the range of selected high-probability mislabeled examples leads to higher accuracy in selecting such examples, ultimately benefiting the classifier's performance.

## 5. Conclusion

In this paper, we focused on the challenge of the inability of existing sample selection methods to effectively select clean hard examples. To address this challenge, we proposed a novel method called *Late Stopping*. In contrast to traditional early-stopping strategies, our method is rooted in a prolonged training process that distinguishes between mislabeled and clean examples by pinpointing the number of training epochs required for each example to be consistently classified to its given label for the first time. We coin this new sample selection criterion as *First-time k-epoch Learning*. Experimental results on synthetic and real-world datasets demonstrate that our proposed method is both straightforward and effective in handling learning with noisy labels. In future work, we plan to enhance the FkL criterion to more accurately capture the intrinsic robust learning ability of DNNs, which could potentially bolster the effectiveness of our *Late Stopping* framework, especially under more intricate noise conditions.

## Acknowledgement

Lei Feng was supported by the National Natural Science Foundation of China (Grant No. 62106028), Chongqing Overseas Chinese Entrepreneurship and Innovation Support Program, CAAI-Huawei MindSpore Open Fund, and Chongqing Artificial Intelligence Innovation Center. Tongliang Liu was partially supported by Australian Research Council Projects IC-190100031, LP-220100527, DP-220102121, and FT-220100318. We would like to thank anonymous reviewers for their constructive feedback that improved our paper.

## References

[1] Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *ICML*, 2017.

[2] Yingbin Bai and Tongliang Liu. Me-momentum: Extracting hard confident examples from noisily labeled data. In *ICCV*, 2021.

[3] Yingbin Bai, Erkun Yang, Bo Han, Yanhua Yang, Jiatong Li, Yinian Mao, Gang Niu, and Tongliang Liu. Understanding and improving early stopping for learning with noisy labels. *NeurIPS*, 2021.

[4] Robert Baldock, Hartmut Maennel, and Behnam Neyshabur. Deep learning through the lens of example difficulty. *NeurIPS*, 2021.

[5] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *ICML*, 2009.

[6] Kaidi Cao, Yining Chen, Junwei Lu, Nikos Arechiga, Adrien Gaidon, and Tengyu Ma. Heteroskedastic and imbalanced deep learning with adaptive regularization. *arXiv*, 2020.

[7] Nicholas Carlini, Ulfar Erlingsson, and Nicolas Papernot. Distribution density, tails, and outliers in machine learning: Metrics and applications. *arXiv*, 2019.

[8] Pengfei Chen, Ben Ben Liao, Guangyong Chen, and Shengyu Zhang. Understanding and utilizing deep neural networks trained with noisy labels. In *ICML*, 2019.

[9] Vitaly Feldman. Does learning require memorization? a short tale about a long tail. In *STOC*, 2020.

[10] Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. *NeurIPS*, 2020.

[11] Guy Hacohen and Daphna Weinshall. On the power of curriculum learning in training deep networks. In *ICML*, 2019.

[12] Bo Han, Gang Niu, Xingrui Yu, Quanming Yao, Miao Xu, Ivor Tsang, and Masashi Sugiyama. Sigua: Forgetting may make learning with noisy labels more robust. In *ICML*, 2020.

[13] Bo Han, Quanming Yao, Tongliang Liu, Gang Niu, Ivor W Tsang, James T Kwok, and Masashi Sugiyama. A survey of label-noise representation learning: Past, present and future. *arXiv*, 2020.

[14] 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. *NeurIPS*, 2018.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[16] Sara Hooker, Aaron Courville, Gregory Clark, Yann Dauphin, and Andrea Frome. What do compressed deep neural networks forget? *arXiv*, 2019.

[17] Wei Hu, Zhiyuan Li, and Dingli Yu. Simple and effective regularization methods for training on noisily labeled data with generalization guarantee. *arXiv*, 2019.

[18] Jinchi Huang, Lie Qu, Rongfei Jia, and Binqiang Zhao. O2u-net: A simple noisy label detection approach for deep neural networks. In *ICCV*, 2019.

[19] Zhuo Huang, Xiaobo Xia, Li Shen, Bo Han, Mingming Gong, Chen Gong, and Tongliang Liu. Harnessing out-of-distribution examples via augmenting content and style. *arXiv*, 2022.

[20] 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 *ICML*, 2018.

[21] Ziheng Jiang, Chiyuan Zhang, Kunal Talwar, and Michael C Mozer. Characterizing structural regularities of labeled data in overparameterized models. In *ICML*, 2021.

[22] Nazmul Karim, Mamshad Nayeem Rizve, Nazanin Rahnavard, Ajmal Mian, and Mubarak Shah. Unicon: Combating label noise through uniform selection and contrastive learning. In *CVPR*, 2022.

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

[24] Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. *arXiv*, 2020.

[25] Mingchen Li, Mahdi Soltanolkotabi, and Samet Oymak. Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. In *AISTATS*, 2020.

[26] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *NeurIPS*, 2020.

[27] Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015.

[28] Yang Liu and Hongyi Guo. Peer loss functions: Learning from noisy labels without knowing noise rates. In *ICML*, 2020.

[29] Bingfeng Luo, Yansong Feng, Zheng Wang, Zhanxing Zhu, Songfang Huang, Rui Yan, and Dongyan Zhao. Learning with noise: Enhance distantly supervised relation extraction with dynamic transition matrix. *arXiv*, 2017.

[30] Yueming Lyu and Ivor W Tsang. Curriculum loss: Robust learning and generalization against label corruption. *arXiv*, 2019.

[31] Pratyush Maini, Saurabh Garg, Zachary Chase Lipton, and J Zico Kolter. Characterizing datapoints via second-split forgetting. In *ICML 2022 Workshop on Spurious Correlations, Invariance and Stability*, 2022.

[32] Karttikeya Mangalam and Vinay Uday Prabhu. Do deep neural networks learn shallow learnable examples first? In *ICML 2019 Workshop on Identifying and Understanding Deep Learning Phenomena*, 2019.

[33] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

[34] Duc Tam Nguyen, Chaithanya Kumar Mummadi, Thi Phuong Nhung Ngo, Thi Hoai Phuong Nguyen, Laura Beggel, and Thomas Brox. Self: Learning to filter noisy labels with self-ensembling. *arXiv*, 2019.

[35] 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 *CVPR*, 2017.

[36] Geoff Pleiss, Tianyi Zhang, Ethan Elenberg, and Kilian Q Weinberger. Identifying mislabeled data using the area under the margin ranking. *NeurIPS*, 2020.

[37] David Rolnick, Andreas Veit, Serge Belongie, and Nir Shavit. Deep learning is robust to massive label noise. *arXiv*, 2017.

[38] Reuven Rubinstein. The cross-entropy method for combinatorial and continuous optimization. *Methodology and computing in applied probability*, 1999.

[39] Yanyao Shen and Sujay Sanghavi. Learning with bad training data via iterative trimmed loss minimization. In *ICML*, 2019.

[40] Hwanjun Song, Minseok Kim, and Jae-Gil Lee. Selfie: Refurbishing unclean samples for robust deep learning. In *ICML*, 2019.

[41] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[42] Daiki Tanaka, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa. Joint optimization framework for learning with noisy labels. In *CVPR*, 2018.

[43] Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J Gordon. An empirical study of example forgetting during deep neural network learning. In *ICLR*, 2018.

[44] Brendan Van Rooyen, Aditya Menon, and Robert C Williamson. Learning with symmetric label noise: The importance of being unhinged. *NeurIPS*, 2015.

[45] Yisen Wang, Weiyang Liu, Xingjun Ma, James Bailey, Hongyuan Zha, Le Song, and Shu-Tao Xia. Iterative learning with open-set noisy labels. In *CVPR*, 2018.

[46] Hongxin Wei, Lei Feng, Xiangyu Chen, and Bo An. Combating noisy labels by agreement: A joint training method with co-regularization. In *CVPR*, 2020.

[47] 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*, 2021.

[48] Qi Wei, Haoliang Sun, Xiankai Lu, and Yilong Yin. Self-filtering: A noise-aware sample selection for label noise with confidence penalization. In *ECCV*, 2022.

[49] Pengxiang Wu, Songzhu Zheng, Mayank Goswami, Dimitris Metaxas, and Chao Chen. A topological filter for learning with label noise. *NeurIPS*, 2020.

[50] 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 *ICCV*, 2021.

[51] Xiaobo Xia, Tongliang Liu, Bo Han, Chen Gong, Nannan Wang, Zongyuan Ge, and Yi Chang. Robust early-learning: Hindering the memorization of noisy labels. In *ICLR*, 2020.

[52] Xiaobo Xia, Tongliang Liu, Bo Han, Mingming Gong, Jun Yu, Gang Niu, and Masashi Sugiyama. Sample selection with uncertainty of losses for learning with noisy labels. *arXiv*, 2021.

[53] 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. *NeurIPS*, 2020.

[54] Yilun Xu, Peng Cao, Yuqing Kong, and Yizhou Wang. L_dmi: A novel information-theoretic loss function for training deep nets robust to label noise. *NeurIPS*, 2019.

[55] Xiangli Yang, Zixing Song, Irwin King, and Zenglin Xu. A survey on deep semi-supervised learning. *IEEE Transactions on Knowledge and Data Engineering*, 2022.

[56] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 2021.

[57] Tianyi Zhou, Shengjie Wang, and Jeff Bilmes. Robust curriculum learning: from clean label detection to noisy label self-correction. In *ICLR*, 2021.