IOFM: Using the Interpolation Technique on the Over-Fitted Models to Identify Clean-Annotated Samples

Dongha Kim[1]*, Yongchan Choi[2]*, Kunwoong Kim[3], Ilsang Ohn[4], Yongdai Kim[3]

[1]Department of Statistics, Sungshin Women's University
[2]Toss bank
[3]Department of Statistics, Seoul National University
[4]Department of Statistics, Inha University
dongha0718@sungshin.ac.kr, {pminer32, kwkim.online, ydkim0903}@gmail.com, ilsang.ohn@inha.ac.kr

Abstract

Most recent state-of-the-art algorithms for handling noisy label problems are based on the *memorization effect*, which is a phenomenon that deep neural networks (DNNs) memorize clean data before noisy ones. While the memorization effect can be a powerful tool, there are several cases where *memorization effect* does not occur. Examples are imbalanced class distributions and heavy contamination on labels. To address this limitation, we introduce a whole new approach called the *interpolation with the over-fitted model (IOFM)*, which leverages over-fitted deep neural networks. The IOFM utilizes a new finding of over-fitted DNNs: for a given training sample, its neighborhoods chosen from the feature space are distributed differently on the original input space depending on the cleanness of the target sample. The IOFM has notable features in two aspects: 1) it yields superior results even when the training data are imbalanced or heavily noisy, 2) since we utilize over-fitted deep neural networks, a fine-tuning procedure to select the optimal training epoch, which is an essential yet sensitive factor for the success of the memorization effect, is not required, and thus, the IOFM can be used for non-experts. Through extensive experiments, we show that our method can serve as a promising alternative to existing solutions dealing with noisy labels, offering improved performance even in challenging and realistic situations.

Introduction

Deep neural networks (DNNs) have achieved impressive successes in many AI tasks but have suffered from collecting massive clean-annotated samples such as ImageNet (Deng et al. 2009) and MS-COCO (Lin et al. 2014). Since annotating procedures are usually done manually by human experts, it is expensive and time-consuming to get extensive clean labeled data, which prevents DNNs from being trained successfully.

On the other hand, it is possible to collect large data easily through internet search engines or hashtags (Fergus et al. 2010; Schroff, Criminisi, and Zisserman 2010; Xiao et al. 2015; Krause et al. 2016). However, the labels are often inaccurate for such data. This has led to increased interest in utilizing datasets with corrupted labels for constructing accurate classifiers, known as the *noisy label problem*.

*These authors contributed equally to this work.

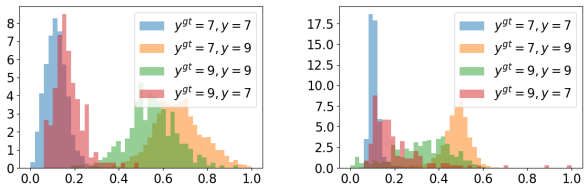

Figure 1: Histograms of the per-sample cross-entropy loss distributions over an imbalanced CIFAR10 dataset with noisy labels at the 1st and 10th epochs. We denote the ground-truth and observed labels by y^{gt} and y, respectively.

Memorization Effect A well-known approach for identifying clean labeled data in the presence of noisy ones is to utilize the *memorization effect* (ME), an interesting characteristic of DNNs. The ME refers to the phenomenon where DNNs tend to memorize clean labeled samples before noisy ones during training (Arpit et al. 2017; Jiang et al. 2018).

With the ME, we can distinguish clean data from contaminated training data by noisy labels (Han et al. 2018). Its simplicity but good performance has inspired numerous follow-up studies, which have achieved great success (Huang et al. 2019; Wu et al. 2020; Mirzasoleiman, Cao, and Leskovec 2020; Pleiss et al. 2020; Cheng et al. 2021; Kim et al. 2021). To the best of our knowledge, there are not many alternatives that can substitute the ME.

Limitation of the Memorization Effect A problem with the ME is that there exist several situations where the strength of the ME diminishes. This occurs, for example, when dealing with imbalanced or heavily contaminated ground-truth label distributions. In such cases, we found that the ME may not appear clearly, leading to suboptimal performance of the follow-up methods in identifying clean labeled data. This is because, in these situations, memorizing clean labeled data first might not be a favorable direction to reducing the overall loss function at early updates.

We provide a simple scenario where the ME does not occur. From CIFAR10, we sample two classes, 7 and 9, and regard the first and second ones as the majority and minority classes, respectively. We gather all images in the first class and only

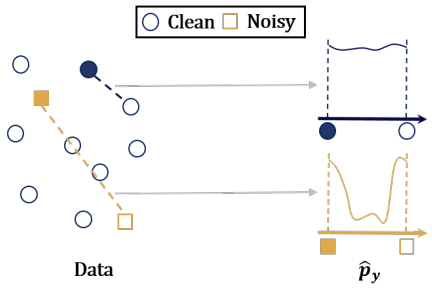

Figure 2: An illustration of the IOFM method. Filled dots represent target inputs, and dashed lines connect them to their nearest neighbor inputs, chosen in the feature space, on the input space. Two graphs present the values of an over-fitted DNN model along each dashed line.

10% randomly sampled images in the second class to make the label distribution be imbalanced. Then for each sample, we flip the class label with probability 0.3 to generate noisy labels. We train a PreActResNet18 (He et al. 2016) using the polluted data and observe the per-sample loss values as the training epoch proceeds, which is depicted in Figure 1.

During the early learning phase, the model consistently memorizes majority data first rather than clean labeled data, which indicates that the ME is not effectively taking place. In this case, considering samples with small losses as clean labeled ones would lead to misidentifying most minor data as noisy labeled.

Overview of Our Method Let us consider an over-fitted DNN and its feature space, i.e., the map of the highest hidden layer. For a given training sample $(\mathbf{x}_*, y_*)$, $\mathbf{x}_*$ is located close to other inputs sharing the label y_* regardless of anomalousness of y_* *on the feature space*. On the other hand, when we consider the *original input space*, the similarity between x_* and its neighbors, chosen from the feature space, would be quite different on the original input space depending on whether y_* is clean or noisy. The similarity becomes smaller when y_* is noisy and vice versa.

Based on this observation of the discrepancy between the similarities on the feature space and the input space, we propose a new and novel method, called *interpolation with the over-fitted model (IOFM)*, for identifying clean labeled samples in a training dataset. Conceptually, The IOFM measures how similar the neighbors of a given datum chosen from the feature space are on the input space and decides the datum as clean when the similarity is large. The visual illustration of our method is depicted in Figure 2. We will explain later how to formalize this idea.

The IOFM has two notable advantages over the existing methods based on the ME. Firstly, our method consistently achieves superior results even in challenging scenarios where memorization-effect-based methods struggle, such as imbalanced or heavily contaminated label distributions. We validate this claim by providing a range of supportive empirical results in the experimental section.

Secondly, our method overcomes numerical instability that other existing ME-based methods often face (Liu et al. 2022a). In particular, the performance of our method does not depend much on the number of updates before making the decision. In contrast, most methods based on the ME require careful tuning of the number of initial updates since the final results depend heavily on this choice. Given that the IOFM delivers robust and excellent performance without requiring the delicate control of various tuning parameters, it serves as a reliable and efficient substitute for ME-based approaches.

This paper is organized as follows. First, we provide a brief review of related studies that address the noisy label problems. Next, we explain detailed descriptions of the IOFM method. Following that, we present the extensive experimental results, including performance tests and ablation studies. Finally, we conclude with closing remarks.

The key contributions of this work are as follows.

- We make a novel observation regarding over-fitted DNNs, specifically the discrepancy between the similarities in the feature space and the original input space for noisy labeled data. Based on this observation, we propose a new method called IOFM.
- Through extensive empirical experiments, we demonstrate that the IOFM method outperforms existing approaches in accurately identifying clean labeled data.
- Additionally, we illustrate that the IOFM can contribute to constructing an accurate classifier in the presence of noisy labeled data.

Related Works

We review related studies that focus on developing efficient algorithms to identify clean data and achieve accurate classifiers by exploiting the ME.

There have been approaches to learn accurate classifiers with noisy labeled training data by loss correction or label correction techniques, pioneered by Patrini et al. (2017); Zhang and Sabuncu (2018). The noise adaptive layer-based algorithm (Goldberger and Ben-Reuven 2017) added additional noisy channels that estimate the correct labels. Wang et al. (2018); Thulasidasan et al. (2019) used the weighted softmax loss function, which is updated based on the current model. There are studies to estimate ground-truth labels directly (Tanaka et al. 2018; Yi and Wu 2019). There is also an attempt to propose a new loss function more robust to noisy labels than standard loss functions (Wang et al. 2018; Thulasidasan et al. 2019; Lyu and Tsang 2020).

Additionally, there is a line of works that ignore the information of noisy labels rather than correcting them. The decouple method (Malach and Shalev-Shwartz 2017) proposes a meta-algorithm called decoupling which decides when to update. D2L (Ma et al. 2018) distinguishes clean labeled data from noisy ones by employing a local dimensionality measure, and ELR (Liu et al. 2020) utilizes the faster gradient vanishings of clean labeled samples at the early learning stage. There are several algorithms to train noisy-robust prediction models by using only a subset of the training data based on their loss or prediction values (Han et al. 2018; Yu et al. 2019; Shen and Sanghavi 2019; Chen et al. 2019; Song, Kim, and Lee 2019; Nguyen et al. 2020). Arazo et al. (2019); Li,

Socher, and Hoi (2020) fit a two-component mixture model on the per-sample loss distribution. Moreover, a couple of works have attempted to develop improved measures compared to using only per-loss values, aiming to fully exploit the ME (Huang et al. 2019; Wu et al. 2020; Mirzasoleiman, Cao, and Leskovec 2020; Pleiss et al. 2020; Cheng et al. 2021; Kim et al. 2021).

Proposed Method

Preliminaries

For a given input vector $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^d$, let $y, y^{\text{gt}} \in [K]$ be its observed and ground-truth labels, respectively, where $[K] = \{1, \ldots, K\}$. We say that the sample $(\mathbf{x}, y)$ is cleanly labeled if $y = y^{\text{gt}}$ and noisily labeled if $y \neq y^{\text{gt}}$. Let $\mathcal{D}^{\text{tr}} = \{(\mathbf{x}_i, y_i), i \in [n]\}$ be a training data set with n samples, and let $\mathcal{C}^{\text{tr}} = \{(\mathbf{x}, y) \in \mathcal{D}^{\text{tr}} : y = y^{\text{gt}}\}$ be the set of clean labeled samples. Our goal is to identify the clean labeled subset $\mathcal{C}^{\text{tr}}$ from $\mathcal{D}^{\text{tr}}$ accurately. Throughout this paper, we abuse the notation $\mathcal{D}^{\text{tr}}$ to denote training *input* data, i.e. $\mathcal{D}^{\text{tr}} = \{\mathbf{x}_i\}_{i=1}^n$, if there is no confusion.

Let $p(\mathbf{x}; \theta) : \mathbb{R}^d \to \mathbb{R}^K$ be a discriminative DNN parametrized by θ which maps an input $\mathbf{x}$ to a K-dimensional conditional probability vector. We denote the k-th component of $p(\mathbf{x}; \theta)$ as $p_k(\mathbf{x}; \theta)$, that is, $p(\mathbf{x}; \theta) = (p_1(\mathbf{x}; \theta), \ldots, p_K(\mathbf{x}; \theta))^\top$. Also, let $h(\mathbf{x}; \theta)$ be the feature vector of $p(\mathbf{x}; \theta)$, the output of the DNN's highest hidden layer. Furthermore, let $p(\mathbf{x}; \widehat{\theta})$ (abbreviated as $\widehat{p}(\mathbf{x})$) be a DNN that perfectly memorizes $\mathcal{D}^{\text{tr}}$ and $h(\mathbf{x}; \widehat{\theta})$ (abbreviated as $\widehat{h}(\mathbf{x})$) be its feature vector.

For a given training sample $(\mathbf{x}_*, y_*) \in \mathcal{D}^{\text{tr}}$, we denote $(\mathbf{x}_{\text{nbd}}, y_{\text{nbd}}) \in \mathcal{D}^{\text{tr}}$ as another training sample that is the nearest to $(\mathbf{x}_*, y_*)$ on the feature space, i.e., $\widehat{h}(\mathcal{X})$, using the Euclidean distance, that is,

$$\mathbf{x}_{\text{nbd}} = \text{argmin}_{\mathbf{x} \in \mathcal{D}^{\text{tr}} \setminus \{\mathbf{x}_*\}} \left\| \widehat{h}(\mathbf{x}) - \widehat{h}(\mathbf{x}_*) \right\|_2 .$$

Motivation: Neighborhood Analysis With an Over-Fitted DNN

We analyze the two-moon dataset, a commonly used synthetic dataset with two classes, depicted in Figure 3-Left. For each data point, we randomly flip its label with probability 0.3 to create the noisy training dataset $\mathcal{D}^{\text{tr}}$. We empirically investigate the different behavior of $\widehat{p}_{y_*}(\mathbf{x})$ depending on the cleanness of a given label y^*.

In Figure 3, we visualize the results. When the training sample $(\mathbf{x}_*, y_*)$ is cleanly labeled, we observe that the nearest neighbor input $\mathbf{x}_{\text{nbd}}$ on the feature space is also located very close to $\mathbf{x}_*$ in the original input space, i.e., $\mathcal{X}$ (depicted by the dot symbols in Figure 3-Left). Thus, $\widehat{p}_{y_*}(\mathbf{x})$ remains large in between $\mathbf{x}_*$ and $\mathbf{x}_{\text{nbd}}$ on the original input space (the upper panel of Figure 3-Right).

Conversely, when y_* is corrupted, the nearest neighbor on the feature space is relatively distant from $\mathbf{x}_*$ in the original input space (indicated by the cross symbols in Figure 3-Left). Hence, it is highly likely that some of clean labeled samples are located between $\mathbf{x}_*$ and $\mathbf{x}_{\text{nbd}}$ in the input space. As a result, there should exist a region between $\mathbf{x}_*$ and $\mathbf{x}_{\text{nbd}}$ in the input space where the value of $\widehat{p}_{y_*}(\mathbf{x})$ is small (the lower panel of Figure 3-Right). We exploit this finding to develop a new score to identify clean labeled data.

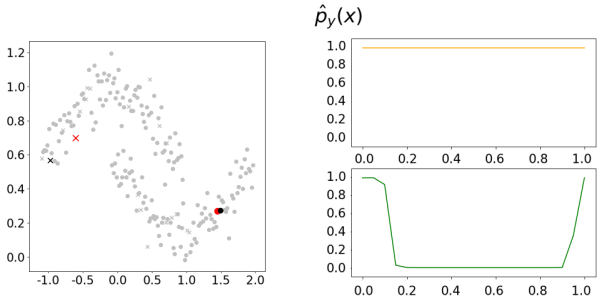

Figure 3: Neighborhood analysis on noisy two-moon. (Left) Scatter plot of the two-moon input data in the input space. Clean and noisy inputs are marked with the dot and cross symbols, respectively. Two target inputs are highlighted in red, and their nearest neighbor inputs chosen in the feature space are in black. (Right) Shapes of $\widehat{p}_{y_*}(\cdot)$ between the two target inputs and their nearest neighbors.

Proposed Algorithm

IOFM Score We propose a new score, the *interpolation with the over-fitted model (IOFM)*, which measures how distant the nearest neighbor of a given datum in the feature space is in the original input space. From the analysis of two-moon, we have noticed a distinct difference in the behavior of $\widehat{p}_{y_*}(\mathbf{x})$ depending on its cleanness, which we will utilize for developing the new score. That is, the area under $\widehat{p}_{y_*}(\mathbf{x})$ over the intervals between $\mathbf{x}_*$ and $\mathbf{x}_{\text{nbd}}$, defined as

$$\int_0^1 \widehat{p}_{y_*}\left(\alpha \mathbf{x}_* + (1-\alpha)\mathbf{x}_{\text{nbd}}\right) d\alpha$$

is large when y_* is clean while it is relatively small when y_* is corrupted. Our proposed score is based on this quantity.

Instead of using the nearest neighbor, it would be, in general, beneficial to use multiple neighbors. Let $\{\mathbf{x}_{\text{nbd},l}\}_{l=1}^L \subset \{\mathbf{x}_i : y_i = y_*, i \in [n]\} \setminus \{\mathbf{x}_*\}$ be L neighborhood training inputs of $\mathbf{x}_*$ in the feature space. Then, we can consider the following averaged score

$$\frac{1}{L}\sum_{l=1}^{L} \int_0^1 \widehat{p}_{y_*}\left(\alpha \mathbf{x}_* + (1-\alpha)\mathbf{x}_{\text{nbd},l})\right) d\alpha. \tag{1}$$

Finally, we approximate the integration in (1) by the trapezoidal rule as follows:

$$s^{\text{IOFM}}(\mathbf{x}_*, y_*) = \frac{1}{L}\sum_{l=1}^{L}\sum_{h=1}^{H} \frac{1}{2H}\left(\widehat{p}_y(\mathbf{x}_{l,h-1}) + \widehat{p}_y(\mathbf{x}_{l,h})\right), \tag{2}$$

where $\mathbf{x}_{l,h} = \frac{H-h}{H}\mathbf{x}_* + \frac{h}{H}\mathbf{x}_{\text{nbd},l}$ and H is the number of trapezoids, to have the IOFM score. A larger value of the IOFM score $s^{\text{IOFM}}(\mathbf{x}_*, y_*)$ indicates more strongly that $(\mathbf{x}_*, y_*)$ is cleanly labeled. In our numerical studies, we set $(L, H) = (10, 3)$ by default, unless otherwise specified.

Theoretical Analysis

For the success of the IOFM, there should exist a DNN having the following three properties: 1) it can memorize the training data well, 2) the model is as smooth as possible between two correctly labeled data in the same ground truth labels and 3) the model changes much between two data whose observed labels are the same but at least one of them is mislabeled. In this section, we show that there does exist a DNN with a reasonable size satisfying these three desirable properties under regularity conditions.

For simplicity, we consider the binary classification problem, i.e., $Y \in \{-1, 1\}$. The extension to multiclass problems can be done without much hamper. Suppose that the training data with ground-truth labels $\{(\mathbf{x}_i, y_i^{\text{gt}}), i \in [n]\}$ are perfectly separable by a smooth decision boundary $g(\mathbf{x})$ with the margin $\gamma > 0$, which can be expressed as follows:

$$\{\mathbf{x}_i \in \mathcal{D}^{\text{tr}} : y_i^{\text{gt}} = 1\} \subset \{\mathbf{x} : g(\mathbf{x}) > \gamma\} \text{ and}$$
$$\{\mathbf{x}_i \in \mathcal{D}^{\text{tr}} : y_i^{\text{gt}} = -1\} \subset \{\mathbf{x} : g(\mathbf{x}) < -\gamma\}.$$

Additionally, let $\mathcal{D}^{\text{tr}}_{s,t} = \{\mathbf{x}_i : y_i = s, y_i^{\text{gt}} = t\}$ for $(s,t) = \{-1,1\}^2$. For any two data points $\mathbf{x}_j$ and $\mathbf{x}_k$, define the line set $L(\mathbf{x}_j, \mathbf{x}_k) = \{\alpha\mathbf{x}_j + (1-\alpha)\mathbf{x}_k, \alpha \in (0,1)\}$. We assume that the line segment between any two correctly labeled data does not cross the decision boundary in the sense that

$$L(\mathbf{x}_j, \mathbf{x}_k) \subset \{\mathbf{x} : g(\mathbf{x}) > \gamma\} \text{ for } \mathbf{x}_j, \mathbf{x}_k \in \mathcal{D}^{\text{tr}}_{1,1}$$
$$L(\mathbf{x}_j, \mathbf{x}_k) \subset \{\mathbf{x} : g(\mathbf{x}) < -\gamma\} \text{ for } \mathbf{x}_j, \mathbf{x}_k \in \mathcal{D}^{\text{tr}}_{-1,-1}$$

Let $d_{jk}(\mathbf{x}_i)$ represent the distance of $\mathbf{x}_i$ from $L(\mathbf{x}_j, \mathbf{x}_k)$, and let $\alpha_n = \min_{j,k} \min_{i \neq j,k} d_{jk}(\mathbf{x}_i)$ and $\beta_n = \min_{j,k} \|\mathbf{x}_j - \mathbf{x}_k\|$. We assume that $\alpha_n > 0$ and $\beta_n > 0$. It is not difficult to show that $\alpha_n = \beta_n = O(n^{-d})$ with probability converging to 1 if $\mathbf{x}_i$s are independent realization of a random vector $\mathbf{X}$ whose distribution has a density.

Given a function $f : \mathcal{X} \to \mathbb{R}$, let $f_1 = f$ and $f_{-1} = -f$. We define $\mathcal{F}(L, r, \tau)$ as the class of DNNs with L many layers, r nodes at each layer, and the sup norm bounded by τ (i.e., $\sup_{f \in \mathcal{F}(L,r,\tau)} \|f\|_\infty \leq \tau$). The following theorem proves the existence of a reasonably sized DNN that satisfies the three desired properties. The proof is in Appendix A.

Theorem 1. *Under the above regularity conditions, there exists a DNN $f \in \mathcal{F}(L, r, \tau)$ for sufficiently large L and r depending on α_n and β_n as well as n such that f is a minimizer of the cross-entropy and satisfies the followings: 1) $y_i f(\mathbf{x}_i) = \tau$ for all $i \in [n]$, 2) $f_s(\mathbf{x})$ is constant on $L(\mathbf{x}_j, \mathbf{x}_k)$ when $\mathbf{x}_j, \mathbf{x}_k \in \mathcal{D}^{tr}_{s,s}$ and 3) $\min_{\mathbf{x} \in L(\mathbf{x}_j, \mathbf{x}_k)} f_s(\mathbf{x}) = -\tau$ whenever $\mathbf{x}_j \in \mathcal{D}^{tr}_{s,-s}$ and $\mathbf{x}_k \in \mathcal{D}^{tr}_{s,s} \cup \mathcal{D}^{tr}_{s,-s}$.*

The success of the IOFM with a DNN in Theorem 1 can be explained as follows. Note that $(\mathbf{x}_*, \mathbf{x}_{\text{nbd},l})$ have the same observed labels (i.e. $y_{\text{nbd}} = y_* = s$). In turn, most $\mathbf{x}_{\text{nbd},l}$ are cleanly labeled since it is assumed that the majority of data is cleanly labeled. Thus, when y_* is clean (i.e. $\mathbf{x}_* \in \mathcal{D}^{\text{tr}}_{s,s}$), most of $\mathbf{x}_{\text{nbd},l}$ are also included in $\mathcal{D}^{\text{tr}}_{s,s}$. Thus, the property 2) of Theorem 1 implies that $\widehat{p}_{y_*}(\mathbf{x})$ would remain large between $\mathbf{x}_*$ and most of $\mathbf{x}_{\text{nbd},l}$, and so the score of the IOFM becomes large. Conversely, when $\mathbf{x}_* \in \mathcal{D}^{\text{tr}}_{s,-s}$, the property 3) of Theorem 1 implies that $\widehat{p}_{y_*}(\mathbf{x})$ becomes negative between $\mathbf{x}_*$ and most of $\mathbf{x}_{\text{nbd},l}$, and so the score of the IOFM becomes smaller relatively to those of clean labeled data.

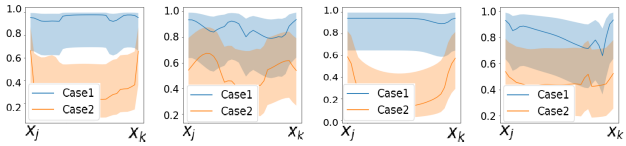

Figure 4: Averaged interpolation results of $\widehat{p}_s(\cdot)$ between two samples on noisy two-moon dataset. (From left to right) We consider four settings regarding imbalanced ratio and noise rate: $(0.5, 0.1)$, $(0.5, 0.3)$, $(0.2, 0.1)$, and $(0.2, 0.3)$.

Algorithm 1: IOFM
In practice, we set $(T_1, T_2, L, H) = (150, 10, 10, 3)$.

input Training data: $\mathcal{D}^{\text{tr}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, a prediction model and its feature function: $p(\cdot;\theta)$ and $h(\cdot;\theta)$, an optimizer $\mathcal{O}$, four integers: T_1, T_2, L, and H.

$\mathcal{S}^{\text{ens}} \leftarrow \emptyset$ // Ensemble IOFM score set
for ($ep = 1$ **to** T_1) **do**
  $MixUp(f(\cdot;\theta), \mathcal{D}^{\text{tr}}, \mathcal{O})$ // train $p(\cdot;\theta)$ using MixUp
  if (ep mod T_2 = 0) **then**
    $\mathcal{S}^{\text{tmp}} \leftarrow \emptyset$
    for ($i = 1$ **to** n) **do**
      $s_i \leftarrow s^{\text{IOFM}}(\mathbf{x}_i, y_i)$ //IOFM score of $(\mathbf{x}_i, y_i)$
      $\mathcal{S}^{\text{tmp}} \leftarrow \text{append}(\mathcal{S}^{\text{tmp}}, s_i)$ // append s_i to $\mathcal{S}^{\text{tmp}}$
    end for
    $\mathcal{S}^{\text{ens}} \leftarrow \mathcal{S}^{\text{ens}} + \mathcal{S}^{\text{tmp}}$
  end if
end for

output $\mathcal{S}^{\text{ens}}$

Remark 2. *While Theorem 1 guarantees the existence of a desirable DNN for the IOFM, it does not ensure its obtainability. Fortunately, there are increasing number of theoretical studies that over-fitted DNNs, trained by minimizing the cross-entropy with the GD algorithm, have a capacity enough to memorize training data while interpolating smoothly between data (Lyu and Li 2020; Chatterji, Long, and Bartlett 2021). Additionally, to demonstrate that such DNNs are indeed achievable, we present interpolation results of $f_s(\mathbf{x})$ between $\mathbf{x}_j$ and $\mathbf{x}_k$. We consider two cases: 1) $\mathbf{x}_j, \mathbf{x}_k \in \mathcal{D}^{tr}_{s,s}$ and 2) $\mathbf{x}_j \in \mathcal{D}^{tr}_{s,-s}$ and $\mathbf{x}_k \in \mathcal{D}^{tr}_{s,s} \cup \mathcal{D}^{tr}_{s,-s}$. We analyze the* `two-moon` *dataset with two ratios of imbalanced samples (0.5&0.2) and two noise ratios (0.1&0.3). For each case, we select ten pairs of $(\mathbf{x}_j, \mathbf{x}_k)$ and plot averaged interpolation along with its 95% confidence band, whose results are depicted in Figure 4.*

Improvement of IOFM

MixUp Loss Function To enhance the smoothness of a trained DNN more while keeping memorizing training data, it would be helpful to use MixUp (Zhang et al. 2017), whose

loss function is given as:

$$\mathop{\mathbb{E}}_{(\mathbf{x}_1,y_1),(\mathbf{x}_2,y_2)\sim\mathcal{D}^{\text{u}}}\mathop{\mathbb{E}}_{\lambda\sim B(\alpha,\alpha)}\text{CE}\left(\text{Mix}_\lambda(y_1,y_2), p(\text{Mix}_\lambda(\mathbf{x}_1,\mathbf{x}_2);\theta)\right), \tag{3}$$

where $\text{CE}(\cdot,\cdot)$ is the cross-entropy, $\text{Mix}_b(u,v) := bu + (1-b)v$, and $B(\alpha,\alpha)$ is the beta distribution with a hyperparameter α. Note that when $\alpha = 0$, the MixUp loss function reduces to the standard cross-entropy loss. Throughout our experiments, we use $B(\alpha,\alpha)$ with $\alpha = 1$, i.e., the uniform distribution. Advantages of using MixUp compared to the standard cross-entropy is well illustrated in Figure 5.

Use of Multiple IOFM Scores We adopt the concept of the AUM (Pleiss et al. 2020) to use multiple IOFM scores obtained from various training epochs. During the training process of the DNN until over-fitting, we calculate the IOFM scores at different training epochs and take the average for the final score. The effectiveness of this ensemble technique will be presented in the experiment section.

Computation Time Reduction in Searching Neighbors The computational cost of calculating neighbors for each training sample can be a concern, especially with large-scale training datasets. This issue can be addressed by only restricting the neighbor search among a small subset of *randomly sampled* training data that share the same label. In practice, we only consider 100 samples as neighborhood candidates and have observed that this approach maintains the performance of the IOFM while significantly reducing the computational time, whose results will be provided in the ablation studies.

IOFM Algorithm Incorporating the above three modifications, the final algorithm of the IOFM method is summarized in Algorithm 1.

Remark 3. *One notable aspect of the IOFM is robustness to the choice of the training epoch. As shown in Figure 5, the accuracy of the IOFM remains stable after reaching its peak around 60 epochs. Considering that other existing methods based on the ME often struggle with the selection of the optimal training epoch, the reliable performance of the IOFM across different epochs boosts the practicability of the IOFM in real data applications, which will be further illustrated in the experiment section.*

Further Extension Towards Constructing Noise-Robust Classifiers

The IOFM can be applied to learn deep classification models with noisy labeled data. In this study, we consider a combination of the IOFM with the DivideMix (Li, Socher, and Hoi 2020), one of the state-of-the-art methods for learning classification models in the presence of noisy labels.

DivideMix decides the cleanness of each datum based on the per-sample loss values, discards the labels of noisy data, and applies a semi-supervised learning algorithm to treat noisy data as unlabeled. Then, it repeat this procedure until accurate prediction models are constructed. To combine the IOMF and DivideMix, we simply substitute the per-sample losses used in the DivideMix with the IOFM scores. A detailed explanation is provided in Appendix B.

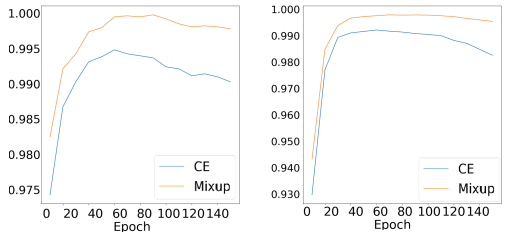

Figure 5: The effect of using the MixUp compared to the standard cross-entropy. We report AUC values for two cases: (Left) 30% and (Right) 50% symmetrically noisy CIFAR10. We train a PreResNet18 (He et al. 2016) for each case.

We want to emphasize that the IOFM can be combined with other ME-based learning frameworks, not just DivideMix. There are many possibilities to explore and further develop improved learning algorithms with noisy labeled data, which would be an interesting future work.

Experiments

Throughout extensive experiments, we empirically evaluate the two aspects of the IOFM. First, we demonstrate the superiority of the IOFM for difficult data such as imbalanced data or heavily noisy cases. Second, we show that the IOFM is really helpful to construct accurate deep classification models in various cases. In each experiment, we report the averaged results based on three trials with random initializations. We use Pytorch framework using a single NVIDIA TITAN XP GPU.

Datasets We provide a brief description of datasets we analyze. The detailed processes of corrupting labels to generate noisy labeled data for each dataset are stated in Appendix C.

First, we analyze two small image datasets, CIFAR10&100 (Krizhevsky and Hinton 2009). Each data set consists of 50K training data and 10K test data with an input size of $3 \times 32 \times 32$, all of which are cleanly labeled. To add noisy labels to CIFAR10&100, we consider both symmetric and asymmetric settings, as done in other studies (Zhang and Sabuncu 2018; Yi and Wu 2019). Additionally, we explore the instance-dependent noise (IDN, Cheng et al. (2021)) and hand-crafted label scenarios (CIFAR10&100-N, Wei et al. (2022b)).

We also analyze two large image datasets, Mini-ImageNet (Vinyals et al. 2016) and Clothing1M (Xiao et al. 2015). We use the subset of the Clothing1M dataset, comprising 48K samples, with roughly 20% noisy labels whose ground-truth labels are known. For Mini-ImageNet, we employ two noisy versions, Blue and Red (Jiang et al. 2020). We note that all results for large datasets are deferred to Appendix D.

Architecture We employ PreActResNet18 (He et al. 2016) for CIFAR10&100, Inception-Resnet-v2 (Szegedy et al. 2017) for Mini-ImageNet, and ResNet50 (Szegedy et al.

Dataset	CIFAR100					
Imbalance type	Step			Long-tail		
Noise type	symm.		Asymm.	symm.		Asymm.
Noise rate (r)	0.3	0.5	0.2	0.3	0.5	0.2
Loss	0.958(0.698)	0.938(0.632)	0.826(0.674)	0.939(0.670)	0.896(0.609)	0.765(0.667)
Ens-Loss	0.969(0.945)	0.951(0.912)	0.829(0.816)	0.954(0.918)	0.915(0.873)	0.796(0.791)
Margin	0.952(0.689)	0.927(0.626)	0.847(0.684)	0.929(0.660)	0.887(0.603)	0.793(0.676)
AUM	0.966(0.851)	0.947(0.779)	0.871(0.802)	0.951(0.798)	0.911(0.727)	0.827(0.775)
sinIOFM	0.965(0.902)	0.949(0.874)	0.862(0.766)	0.953(0.872)	0.909(0.833)	0.826(0.723)
IOFM	**0.973(0.958)**	**0.958(0.936)**	**0.911(0.910)**	**0.961(0.942)**	**0.927(0.905)**	**0.875(0.874)**

Table 1: Comparison of the clean/noisy classification AUC values on the imbalanced CIFAR100. We list the best and final (in the parentheses) results.

Dataset	CIFAR10			CIFAR100		
Noise type	Symm.		Asymm.	Symm.		Asymm.
Noise rate (r)	0.8	0.9	0.4	0.8	0.9	0.4
Loss	0.919(0.569)	0.836(0.556)	0.933(0.753)	0.847(0.557)	0.708(0.521)	0.621(0.552)
Ens-Loss	0.947(0.895)	0.867(0.763)	0.908(0.901)	0.873(0.704)	0.733(0.660)	0.642(0.638)
Margin	0.918(0.567)	0.834(0.554)	**0.940**(0.756)	0.833(0.553)	0.704(0.520)	0.634(0.553)
AUM	0.948(0.809)	0.869(0.687)	0.927(0.874)	0.874(0.693)	0.734(0.588)	0.677(0.637)
sinIOFM	0.924(0.699)	0.842(0.622)	0.912(0.825)	0.859(0.684)	0.714(0.567)	0.664(0.607)
IOFM	**0.954(0.907)**	**0.887(0.811)**	0.934(**0.921**)	**0.890(0.806)**	**0.746(0.653)**	**0.713(0.712)**

Table 2: Clean/noisy classification AUC results on heavily noisy CIFAR10&100. The best and final (in the parentheses) results are listed.

2015) for Clothing1M, respectively. For the latter two architectures, we utilize pre-trained models trained on ImageNet. Details about the learning schedules can be found in Appendix C.

Clean Data Identification Performance

We begin by assessing the accuracy of the IOFM in distinguishing clean labeled data from noisy ones and compare it with other baseline methods. We assess the performance using the clean/noisy classification AUC values on the training data. Our focus is on two challenging scenarios: 1) datasets with imbalanced ground-truth label distributions, and 2) datasets with highly corrupted labels.

We consider two versions of the IOFM: 1) the original IOFM (IOFM) based on the ensemble of multiple IOFM scores from multiple epochs, and 2) the IOFM based on the single score at the last epoch (sinIOFM). We consider sinIOFM since it is computationally simpler than the IOFM.

For baselines, we consider two methods based on the ME: 1) small-loss method (Loss), a frequently adopted strategy in numerous studies that uses the per-sample-loss as the score, and 2) AUM (Pleiss et al. 2020). We also include the ensembled version of the small-loss method (Ens-loss), using averaged per-loss values from multiple epochs, as well as the AUM without ensemble (Margin).

Imbalanced Case The original image datasets have balanced labels. To create imbalanced data, we employ two subsampling strategies: 1) the Step strategy and 2) the Long-tail strategy, which are considered in Cao et al. (2019). Detailed descriptions of these strategies are stated in Appendix C.

Table 1 presents the results for CIFAR100. The results for CIFAR10 and Mini-ImageNet datasets, which are similar, are included in Appendix D. In all cases, the IOFM dominates all of the baselines as well as sinIOFM. Additionally, sinIOFM is superior to Loss and Margin which do not use ensemble techniques, while being competitive to Loss-Ens and AUM both of which employ ensemble techniques. These observations imply that ensemble techniques are generally helpful for noisy label detection, despite their increased computations, and sinIFOM is a useful alternative to the IOFM when computing resource for the ensemble technique is not sufficient.

Another interesting observation is that the performance gap between the best and last results for the IOFM is minimal. Even the last results of the IOFM are favorably comparable with the best results of the baselines. This suggests that the IOFM can be readily applied in practice without fine-tuning the optimal epoch. In contrast, the optimal tuning of the epoch is indispensable for the baselines which would be difficult and thus hamper their applications to real data analysis.

Heavily Noisy Case Table 2 summarizes the results on the heavily noised case on CIFAR10&100. Similar to the imbalanced scenario, the IOFM performs well to achieve the best or the second best performance for the all cases. Furthermore, the differences of the best and last results of the IOFM are the smallest, which means that the IOFM can be implemented in practice without fine-tuning. We believe that the IOFM could be an off-the-shelf algorithm for noisy label detection.

Dataset	CIFAR10			
Imbalance type	Step			
Noise type	symm.		Asymm.	
Noise rate (r)	0.3	0.5	0.2	0.4
Cross-Entropy	72.72	68.99	81.19	74.95
Mixup	74.81	65.63	81.97	75.93
DivideMix	85.76	85.16	87.15	78.35
IOFM+DivideMix	**88.19**	**88.07**	**87.35**	**78.73**

Table 3: Comparison of the best test accuracies(%) of various methods on the imbalanced CIFAR10.

Data set	CIFAR10		
Noise type	Symm.		Asymm.
Noise rate (r)	0.8	0.9	0.4
Cross-Entropy	62.9	42.7	85.0
Co-teaching+ (Yu et al. 2019)	67.4	47.9	-
P-correction (Yi and Wu 2019)	77.5	58.9	88.5
MLNT (Li et al. 2019)	-	59.1	89.2
M-correction (Arazo et al. 2019)	86.8	69.1	87.4
PENCIL (Yi and Wu 2019)	77.5	58.9	88.5
DivideMix	92.90	71.34	93.36
ELR+ (Liu et al. 2020)	93.3	78.7	93.0
IOFM+DivideMix	**93.48**	**81.20**	**93.41**

Table 4: Comparison of the best test accuracies(%) of various methods on highly corrupted CIFAR10. The results of DivideMix are re-implemented by us.

Classification Accuracy Performance

We carry out test accuracy comparison of the modified DivideMix with the IOFM (IOFM+DivideMix) with other baseline methods under various scenarios, including those considered in the previous section.[1] We note that additional experimental results, such as those in IDN scenarios, can be found in Appendix D.

Table 3 shows the test accuracies of the best prediction models for each method on the imbalanced CIFAR10 with the step strategy. The results of the imbalanced CIFAR10 with the long-tail strategy and the imbalanced CIFAR100 with both strategies can be found in Appendix D. We can clearly observe that the utilization of IOFM scores enhances the original DivideMix in most cases and consistently dominates the other approaches. Specifically, in situations with symmetric noise, our method often outperforms the original DivideMix by over 3%.

The results for heavily noisy cases are presented in Table 4. For existing methods, we include the prediction accuracies given in their respective papers. Similar to the imbalanced cases, the IOFM consistently improves DivideMix in all scenarios and outperforms other state-of-the-art methods, indicating the effectiveness of using IOFM to improve exiting algorithms. Notably, in the case where 90% of labels are corrupted in CIFAR10, our method demonstrates high performance, enhancing DivideMix by nearly 10%.

[1]Our source code is based on the publicly available GitHub code of DivideMix.

Dataset	CIFAR10-N		
Noise type	Aggre	Rand1	Worst
ERL+ (Liu et al. 2020)	94.83	94.43	91.09
CORES (Cheng et al. 2021)	95.25	94.45	91.66
NLS (Wei et al. 2022a)	91.97	90.29	82.99
SOP+ (Liu et al. 2022b)	95.61	95.28	**93.24**
ROBOT (Lin et al. 2023)	91.35	90.46	84.05
IOFM+DivideMix	**95.71**	**95.62**	93.04

Table 5: Comparison of the best test accuracies(%) of various methods on CIFAR10-N.

We also explore CIFAR10&100 with hand-crafted annotations, CIFAR10&100-N. The results for CIFAR10-N are summarized in Table 5 and the results for CIFAR100-N are in Appendix D. We present results over three noisy scenarios. Our method consistently achieves the best or second-best accuracies compared to other recent baselines in all cases.

Ablation Studies

We conduct additional experiments to gain further insights for the IOFM. The summarized results of these experiments are provided in In the following, we provide the summary of further experiments, whose detailed results including tables and figures are deferred to Appendix D. 1) Our method becomes more accurate as the value of L increases, but it saturates when $L \geq 80$. 2) $H = 3$ is sufficient for approximating the IOFM score. 3) For an imbalanced CIFAR10, the neighborhood search using sampling takes only 2.13 seconds, which is about 50 times faster compared to the non-sampling case. Considering that training a model with the CE and MixUp for a single epoch require 10.56 and 10.87 seconds, respectively, the sampling strategy makes the total running time of the IOFM comparable to other existing methods. 4) The sampling method maintains the performance of the IOFM. 5) The IOFM is robust to choosing models and optimizers.

Concluding Remarks

In this study, we have developed a novel approach named the IOFM for identifying clean labeled samples within training data that contain noisy labels. Our approach is based on a novel finding that there is a discordance between noisy and clean labeled data with respect to the distribution of the prediction values of an over-fitted DNN around the neighborhood on the feature space. Combined with MixUp loss function, incorporating multiple scores, we empirically demonstrated that the IOFM achieves superior results, particularly in challenging and realistic scenarios, and does not suffer from fine-tuning the optimal choice of training epoch.

It would be interesting to apply our methods to supervised anomaly detection tasks (Pang et al. 2021). When the information about the anomalousness (normal or abnormal) of each training datum is available but not entirely accurate, we can regard the task as the noisy label problem with an extremely imbalanced label distribution. We expect that our methods would successfully address this anomaly detection problem.

Acknowledgements

DK was supported by the Sungshin Women's University Research Grant of H20230044. YK was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2020R1A2C3A01003550) and Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) [NO.2022-0-00184, Development and Study of AI Technologies to Inexpensively Conform to Evolving Policy on Ethics]. IO was supported by INHA UNIVERSITY Research Grant.

References

Arazo, E.; Ortego, D.; Albert, P.; O'Connor, N. E.; and McGuinness, K. 2019. Unsupervised Label Noise Modeling and Loss Correction. In *36th International Conference on Machine Learning*, 312–321.

Arpit, D.; Jastrzkebski, S.; Ballas, N.; Krueger, D.; Bengio, E.; Kanwal, M. S.; Maharaj, T.; Fischer, A.; Courville, A.; Bengio, Y.; et al. 2017. A closer look at memorization in deep networks. In *International Conference on Machine Learning*, 233–242. PMLR.

Cao, K.; Wei, C.; Gaidon, A.; Arechiga, N.; and Ma, T. 2019. Learning imbalanced datasets with label-distribution-aware margin loss. *Advances in neural information processing systems*, 32.

Chatterji, N. S.; Long, P. M.; and Bartlett, P. 2021. When does gradient descent with logistic loss interpolate using deep networks with smoothed ReLU activations? In *Conference on Learning Theory*, 927–1027. PMLR.

Chen, P.; Liao, B. B.; Chen, G.; and Zhang, S. 2019. Understanding and Utilizing Deep Neural Networks Trained with Noisy Labels. In Chaudhuri, K.; and Salakhutdinov, R., eds., *Proceedings of the 36th International Conference on Machine Learning*, 1062–1070. PMLR.

Cheng, H.; Zhu, Z.; Li, X.; Gong, Y.; Sun, X.; and Liu, Y. 2021. Learning with Instance-Dependent Label Noise: A Sample Sieve Approach. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.

Fergus, R.; Fei-Fei, L.; Perona, P.; and Zisserman, A. 2010. Learning object categories from internet image searches. *Proceedings of the IEEE*, 98(8): 1453–1466.

Goldberger, J.; and Ben-Reuven, E. 2017. Training deep neural-networks using a noise adaptation layer. In *5th International Conference on Learning Representations*.

Han, B.; Yao, Q.; Yu, X.; Niu, G.; Xu, M.; Hu, W.; Tsang, I. W.; and Sugiyama, M. 2018. Co-Teaching: Robust Training of Deep Neural Networks with Extremely Noisy Labels. In *32nd International Conference on Neural Information Processing Systems*, 8536–8546.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Identity Mappings in Deep Residual Networks. In *European Conference on Computer Vision*, 630–645. Springer.

Huang, J.; Qu, L.; Jia, R.; and Zhao, B. 2019. O2u-net: A simple noisy label detection approach for deep neural networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, 3326–3334.

Jiang, L.; Huang, D.; Liu, M.; and Yang, W. 2020. Beyond synthetic noise: Deep learning on controlled noisy labels. In *International conference on machine learning*, 4804–4815. PMLR.

Jiang, L.; Zhou, Z.; Leung, T.; Li, L.-J.; and Fei-Fei, L. 2018. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International Conference on Machine Learning*, 2304–2313. PMLR.

Kim, T.; Ko, J.; Choi, J.; Yun, S.-Y.; et al. 2021. Fine samples for learning with noisy labels. *Advances in Neural Information Processing Systems*, 34: 24137–24149.

Krause, J.; Sapp, B.; Howard, A.; Zhou, H.; Toshev, A.; Duerig, T.; Philbin, J.; and Fei-Fei, L. 2016. The unreasonable effectiveness of noisy data for fine-grained recognition. In *European Conference on Computer Vision*, 301–320. Springer.

Krizhevsky, A.; and Hinton, G. 2009. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*.

Li, J.; Socher, R.; and Hoi, S. C. 2020. Dividemix: Learning with noisy labels as semi-supervised learning. In *8th International Conference on Learning Representations*.

Li, J.; Wong, Y.; Zhao, Q.; and Kankanhalli, M. S. 2019. Learning to learn from noisy labeled data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5051–5059.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 740–755.

Lin, Y.; Pi, R.; Zhang, W.; Xia, X.; Gao, J.; Zhou, X.; Liu, T.; and Han, B. 2023. A Holistic View of Label Noise Transition Matrix in Deep Learning and Beyond. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Liu, S.; Niles-Weed, J.; Razavian, N.; and Fernandez-Granda, C. 2020. Early-Learning Regularization Prevents Memorization of Noisy Labels. In *Advances in Neural Information Processing Systems 33 pre-proceedings (NeurIPS 2020)*.

Liu, S.; Zhu, Z.; Qu, Q.; and You, C. 2022a. Robust training under label noise by over-parameterization. In *International Conference on Machine Learning*, 14153–14172. PMLR.

Liu, S.; Zhu, Z.; Qu, Q.; and You, C. 2022b. Robust Training under Label Noise by Over-parameterization. In Chaudhuri, K.; Jegelka, S.; Song, L.; Szepesvári, C.; Niu, G.; and Sabato, S., eds., *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, 14153–14172. PMLR.

Lyu, K.; and Li, J. 2020. Gradient Descent Maximizes the Margin of Homogeneous Neural Networks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Lyu, Y.; and Tsang, I. W. 2020. Curriculum Loss: Robust Learning and Generalization against Label Corruption. In *8th International Conference on Learning Representations*.

Ma, X.; Wang, Y.; Houle, M. E.; Zhou, S.; Erfani, S.; Xia, S.; Wijewickrema, S.; and Bailey, J. 2018. Dimensionality-Driven Learning with Noisy Labels. In *35th International Conference on Machine Learning*, 5907–5915.

Malach, E.; and Shalev-Shwartz, S. 2017. Decoupling "when to update" from "how to update". In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, 960–970. Curran Associates, Inc.

Mirzasoleiman, B.; Cao, K.; and Leskovec, J. 2020. Coresets for robust training of deep neural networks against noisy labels. *Advances in Neural Information Processing Systems*, 33: 11465–11477.

Nguyen, D. T.; Mummadi, C. K.; Ngo, T. P. N.; Nguyen, T. H. P.; Beggel, L.; and Brox, T. 2020. SELF: Learning to Filter Noisy Labels with Self-Ensembling. In *International Conference on Learning Representations*.

Pang, G.; Shen, C.; Cao, L.; and Hengel, A. V. D. 2021. Deep learning for anomaly detection: A review. *ACM computing surveys (CSUR)*, 54(2): 1–38.

Patrini, G.; Rozza, A.; Krishna Menon, A.; Nock, R.; and Qu, L. 2017. Making Deep Neural Networks Robust to Label Noise: A Loss Correction Approach. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2233–2241.

Pleiss, G.; Zhang, T.; Elenberg, E.; and Weinberger, K. Q. 2020. Identifying mislabeled data using the area under the margin ranking. *Advances in Neural Information Processing Systems*, 33: 17044–17056.

Schroff, F.; Criminisi, A.; and Zisserman, A. 2010. Harvesting Image Databases from the Web. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(4): 754–766.

Shen, Y.; and Sanghavi, S. 2019. Learning with Bad Training Data via Iterative Trimmed Loss Minimization. In Chaudhuri, K.; and Salakhutdinov, R., eds., *36th International Conference on Machine Learning*, 5739–5748. PMLR.

Song, H.; Kim, M.; and Lee, J.-G. 2019. SELFIE: Refurbishing Unclean Samples for Robust Deep Learning. 5907–5915. PMLR.

Szegedy, C.; Ioffe, S.; Vanhoucke, V.; and Alemi, A. 2017. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; and Rabinovich, A. 2015. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1–9.

Tanaka, D.; Ikami, D.; Yamasaki, T.; and Aizawa, K. 2018. Joint optimization framework for learning with noisy labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5552–5560.

Thulasidasan, S.; Bhattacharya, T.; Bilmes, J.; Chennupati, G.; and Mohd-Yusof, J. 2019. Combating Label Noise in Deep Learning Using Abstention. *Proceedings of the 36th International Conference on Machine Learning*, 6234–6243.

Vinyals, O.; Blundell, C.; Lillicrap, T.; Wierstra, D.; et al. 2016. Matching networks for one shot learning. *Advances in neural information processing systems*, 29.

Wang, Y.; Liu, W.; Ma, X.; Bailey, J.; Zha, H.; Song, L.; and Xia, S.-T. 2018. Iterative Learning with Open-set Noisy Labels. In *2018 IEEE Conference on Computer Vision and Pattern Recognition*, 8688–8696.

Wei, J.; Liu, H.; Liu, T.; Niu, G.; Sugiyama, M.; and Liu, Y. 2022a. To Smooth or Not? When Label Smoothing Meets Noisy Labels. In Chaudhuri, K.; Jegelka, S.; Song, L.; Szepesvári, C.; Niu, G.; and Sabato, S., eds., *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, 23589–23614. PMLR.

Wei, J.; Zhu, Z.; Cheng, H.; Liu, T.; Niu, G.; and Liu, Y. 2022b. Learning with Noisy Labels Revisited: A Study Using Real-World Human Annotations. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Wu, P.; Zheng, S.; Goswami, M.; Metaxas, D.; and Chen, C. 2020. A topological filter for learning with label noise. *Advances in neural information processing systems*, 33: 21382–21393.

Xiao, T.; Xia, T.; Yang, Y.; Huang, C.; and Wang, X. 2015. Learning from massive noisy labeled data for image classification. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2691–2699.

Yi, K.; and Wu, J. 2019. Probabilistic end-to-end noise correction for learning with noisy labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7017–7025.

Yu, X.; Han, B.; Yao, J.; Niu, G.; Tsang, I.; and Sugiyama, M. 2019. How does Disagreement Help Generalization against Label Corruption? In Chaudhuri, K.; and Salakhutdinov, R., eds., *36th International Conference on Machine Learning*, 7164–7173. PMLR.

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2017. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.

Zhang, Z.; and Sabuncu, M. 2018. Generalized cross entropy loss for training deep neural networks with noisy labels. In *32nd International Conference on Neural Information Processing Systems*, 8792–8802.