ROBUST LEARNING WITH ADAPTIVE SAMPLE CREDIBILITY MODELING

Anonymous authors
Paper under double-blind review

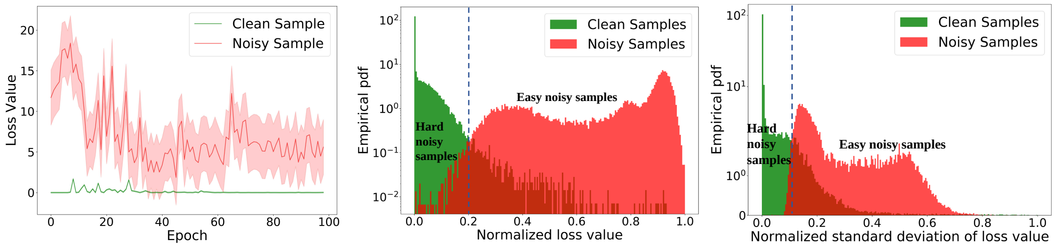
ABSTRACT

Training deep neural network (DNN) with noisy labels is practically challenging since inaccurate labels severely degrade the generalization ability of DNN. Previous efforts tend to handle part or full data in a unified denoising flow to mitigate the noisy label problem, while they lack the consideration of intrinsic difference among difficulties of various noisy samples. In this paper, a novel and adaptive end-to-end robust learning method, called *CREMA*, is proposed. The insight behind is that the credibility of a training sample can be estimated by the joint distribution of its data-label pair, thus to roughly separate clean and noisy samples from original samples, which will be processed with different denoising process in a divide-and-conquer manner. For the clean set, we deliberately design a memory-based modulation scheme to dynamically adjust the contribution of each sample in terms of its historical credibility sequence during training, thus to alleviate the effect from potential hard noisy samples in clean set. Meanwhile, for those samples categorized into noisy set, we try to correct their labels in a selective manner to maximize data utilization and further boost performance. Extensive experiments on mainstream benchmarks, including synthetic (noisy versions of MNIST, CIFAR-10 and CIFAR-100) and real-world (Clothing1M and Animal-10N) noisy datasets demonstrate superiority of the proposed method.

1 INTRODUCTION

Deep learning has achieved significant progress in the field of computer vision and language processing. The key to its success is the availability of large scale dataset with reliable annotations. Collecting such dataset, however, is time-consuming and expensive. Easy ways to obtain labeled data, such as web crawling (Xiao et al., 2015a), inevitably yield samples with noisy label, which is not apporiate to be directly utilized to train DNN since these complex models can easily memorizing noisy labels (Arpit et al., 2017; Zhang et al., 2017).

To handle this problem, classical Learning with Noisy Label (LNL) approaches focus on either identifying and dropping noisy samples (i.e., sample selection) (Han et al., 2018b; Jiang et al., 2018) or adjusting the objective term of each sample during training (i.e., loss adjustment) (Patrini et al., 2017; Yi & Wu, 2019). The former usually make use of small-loss trick to select clean samples, and then take them to update DNNs. However, the procedure of sample selection cannot guarantee that the selected clean samples are completely clean. In contrast, as indicated in Fig. 1, division relied on statistic metric can still involve some hard noisy samples into training set, which will be treated equally as other normal samples in following training stages. Thus the negative impact brought by wrongly grouped noisy samples can still confuse the optimization process and lower the test performance of DNNs (Yu et al., 2019). On the other hand, the latter schemes reweight loss values or update labels by estimating the confidence of each sample being clean. Typical methods include loss correction via estimated noise transition matrix (a small set of clean samples are usually required to obtain more accurate noise transition matrix) (Hendrycks et al., 2018). However, estimating accurate noise transition matrix is still challenging and the assumption of availability of a small clean dataset cannot be fulfilled in many real-world scenarios. Recently, there are approaches directly correcting the labels of all training samples (Tanaka et al., 2018; Yi & Wu, 2019). However, we empirically find that unconstrained label correction in full data can do harm to clean samples and reversely hinder the model performance.



(a) (b) (c)

Figure 1: Training on MNIST with 50% symmetric noise. (a) Compared with noisy samples, clean samples yield relatively smaller loss value and more consistent prediction. (b) Empirical pdf of loss values and (c) their stand deviation justify above conclusion. That is, clean and noisy samples possess distinctive statistical properties. However, noisy samples can not be completely identified via a simple threshold filter strategy (blue dotted line in (b) and (c)) with these statistical metrics. The existence of easy and hard noisy samples requiring different ways to handle them accordingly.

Towards the problems above, we propose a simple but effective method called *CREMA* (*sample CREdibility Modeling and Adaptive loss reweighting*), which adaptively reduces the impact of noisy labels via modeling the credibility (i.e., quality) of each sample. With the estimated sample credibility, clean and noisy samples can be separated and handled in a divide-and-conquer manner. Since it is practically impossible to separate these samples perfectly, for the selected clean samples, we take their historical credibility sequences to adjust contribution of each sample to their objective, thus to mitigate the negative impact of hard noisy samples. As for the separated noisy samples, some of them are actually clean (i.e., hard samples) and can be useful. Thus instead of discarding them as in previous sample selection methods (Han et al., 2018b; Wei et al., 2020), we make use of them via a selective label correction scheme.

Our insight is from the observation on the loss value during training on noisy data (illustrated in Fig 1), it can be found that **clean and noisy samples manifest distinctive statistical properties during training, where clean samples yield relatively smaller loss value** (Reed et al., 2015) **and more consistent prediction**. Hence these statistical features can be utilized to model the sample credibility. However, Fig 1 also shows that the full data can not be perfectly separated by simple statistical metrics. This inspires us to adaptively cope with noises of different difficulty levels. For easily recognized noisy samples, we can directly apply certain label correction scheme while avoiding erroneous correction on normal samples. For samples fall into the confused area and hybrid with clean ones, since estimated credibility in current epoch is not informative enough to identify noisy samples, *CREMA* resorts to a historical sequence of sample credibility to infer the likelihood of being noisy data-label pair. This is implemented by maintaining a historical memory-bank along with the training process and estimating the likelihood function through a consistency metric and assumption of markov property of the sequence.

CREMA is built upon a classic co-training framework (Wei et al., 2020; Han et al., 2018b). The sample credibility estimated by one network is used to adjust the loss term of credible samples for the other network. Extensive experiments conducted on mainstream benchmarks, including synthetic (noisy versions of MNIST, CIFAR-10 and CIFAR-100) and real-world (Clothing1M and Animal-10N) noisy datasets demonstrate superiority of the proposed method. In a nutshell, the key contributions of this paper include:

• *CREMA*: a novel LNL algorithm that combating noisy labels via separating clean and noisy samples and making use of them respectively, in spirit of the idea of divide-and-conquer. Easily recognized noisy samples are handled via a selective label update strategy;

• In *CREMA*, likelihood estimation of historical credibility sequence is proposed to help distinguish hard noisy samples, which naturally plays as the dynamical weight to modulate loss term of each training sample;

• *CREMA* is tested on five synthetic and real-world noisy datasets across various types and levels of label noise. Extensive ablation studies and qualitative analysis are provided to verify the effectiveness of each component.

The source code and supporting materials of our work will be published online upon acceptance.

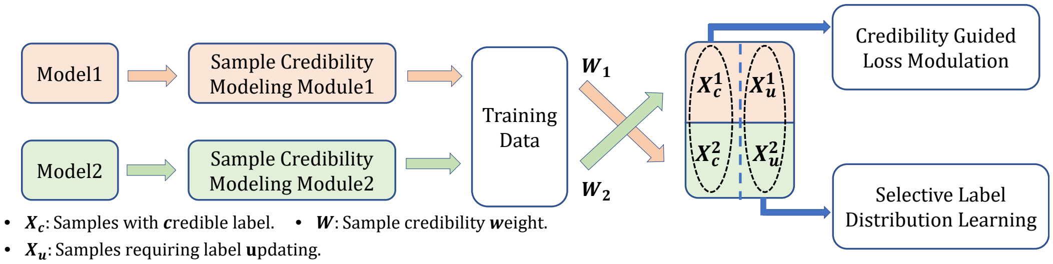

Model1
Sample Credibility Modeling Module1
Model2
Sample Credibility Modeling Module2
Training Data
W_1
W_2
X_c^1
X_u^1
X_c^2
X_u^2
Credibility Guided Loss Modulation
Selective Label Distribution Learning
• X_c: Samples with credible label.
• W: Sample credibility weight.
• X_u: Samples requiring label updating.

Figure 2: The pipeline of *CREMA*. *CREMA* trains two parallel networks simultaneously. Clean samples (mostly clean) X_c and noisy samples (mostly noisy) X_u are separated via estimating the credibility of each training data. A selective label distribution learning scheme is applied for easily distinguishable noisy samples in X_u. As for the clean set X_c, likelihood estimation of historical credibility sequence is proposed to handle the hard noisy samples via adaptively modulate their loss term during training.

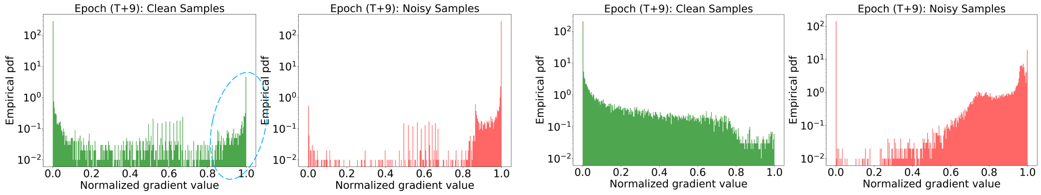
2 Related Works

The existing LNL approaches can be mainly categorized into three groups: loss adjustment, label correction and noisy sample detection. Next we will introduce and discuss existing works for training DNN with noisy labels.

Loss Adjustment. Adjusting the loss values of all training samples is able to reduce the negative impact of noisy labels. To do this, many approaches seek to robust loss function, such as Robust MAE (Ghosh et al., 2017), generalized cross entropy (Zhang & Sabuncu, 2018), symmetric cross entropy (Wang et al., 2019) and curriculum loss (Lyu & Tsang, 2020). Rather than treat all samples equally, some methods rectify the loss of each sample through estimating label transition matrix (Hendrycks et al., 2018; Patrini et al., 2017; Goldberger & Ben-Reuven, 2017; Han et al., 2018a; Xiao et al., 2015b) or imposing different importance to each sample to formulate a weighted training procedure (Wang et al., 2017; Liu & Tao, 2015; Chang et al., 2017). The noise transition matrix, however, is relatively hard to be estimated and many approaches (Hendrycks et al., 2018; Veit et al., 2017; Litany & Freedman, 2018; Zhang et al., 2020b; Jiang et al., 2018; Li et al., 2017; Dehghani et al., 2017b;a; Ren et al., 2018; Shu et al., 2019; Zhang et al., 2020b;a) make assumptions that a small clean-labeled dataset exists. In real-world scenarios, such condition is not always fulfilled, thus limiting the applications of these approaches.

Label correction. Label correction methods seek to refurbish the ground-truth of noisy samples, thus preventing DNN overfits to false labels. The most common ways to obtain the updated label includes bootstrapping (i.e., a convex combination of the noisy label and the DNN prediction) (Reed et al., 2015; Han et al., 2019; Arazo et al., 2019) and label replacing (Tanaka et al., 2018; Yi & Wu, 2019; Song et al., 2019; Zhang et al., 2021). One critical problem of label correction methods is to define the confidence of each label being clean, that is, samples with high clean probability should keep their labels almost unchanged, and vice versa. Previous solutions including cross-validation (Reed et al., 2015), fitting a two-component mixture model (Arazo et al., 2019), local intrinsic dimensionality measurement (Houle, 2017; Ma et al., 2018) and leveraging the prediction consistency of DNN models (Song et al., 2019). However, updating the labels of all training set is challenging, and well-designed regularization terms are important to prevent DNN from falling into trivial solutions (Tanaka et al., 2018; Yi & Wu, 2019).

Noisy sample detection. One commonly knowledge used to discover noisy samples is the *memorization effects* (i.e., DNN fits clean samples first and then noisy ones). As a result, after a warm-up training stage with all noisy samples, DNN is able to identify the clean samples by taking the small-loss ones. The *small-loss trick* is exploited by many sample selection methods (Han et al., 2018b; Yu et al., 2019; Malach & Shalev-Shwartz, 2017; Jiang et al., 2018; Wei et al., 2020; Nguyen et al., 2020; Song et al., 2019; Yao et al., 2020). After separating the noisy samples from the clean ones, Co-teaching (Han et al., 2018b) and it's many variants (Han et al., 2018b; Yu et al., 2019; Wei et al., 2020; Yao et al., 2020) updates two parallel network's parameters with the clean samples and abandoned the noisy ones. The idea of training two deep networks simultaneously is effective to avoid



(a) Global label learning.

(b) Selective label learning in *CREMA*.

Figure 3: Training on MNIST with 50% symmetric noise, warm up (i.e., training on all samples with original noisy labels) for T epochs. (a) Global learning procedure requires updating all training samples' label, causes relatively large gradient values even on clean samples (areas within blue dotted lines), making it hard to focus on correcting noisy labels. (b) Training on *CREMA* can effectively identify noisy samples and focus on correcting noisy labels with reletively large gradient values.

the confirmation bias problem (i.e., a model would accumulate its error trough the self-training process) (Han et al., 2018b; Jiang et al., 2018; Li et al., 2020). Discarding the noisy samples however, means that valuable data may be lost, which leads to slow convergence of DNN models (Chang et al., 2017). Instead, there are methods that utilize both clean and noisy samples to formulate a semi-supervised learning problem, by discarding only the labels of noisy samples. Thus converting LNL problem into a semi-supervised learning ones, for which powerful semi-supervised learning methods can be leveraged to boost performance (Li et al., 2020; Zheltonozhskii et al., 2021; Zhang et al., 2018; Berthelot et al., 2019).

The proposed method *CREMA*, is a hybrid method that takes advantages of loss adjustment, label correction and sample selection. As shown in Fig. 2, *CREMA* splits training data into clean and noisy set and proceeds a divide-and-conquer strategy. Label correction scheme is utilized to handle the easily recognizable noisy samples, while for hard noisy ones, loss adjustment plays important roles to modulate contribution of each sample via a novel sample credibility modeling method.

3 Method

In this section, we introduce *CREMA*, an end-to-end approach for LNL problem. The pipeline of the approach is shown in Fig. 2. Our training process is built upon a classic co-training framework (Han et al., 2018b; Wei et al., 2020) to avoid confirmation bias and separate credible samples (mostly clean) and noisy samples (mostly noisy) via per-sample loss values.

Formally, for multi-class classification problem, let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ denote the training data, where x_i is an image and $y_i \in \{0,1\}^C$ is the one-hot label over C classes. $f(x_i;\theta)$ denotes image feature extracted by DNN model. With the loss from the DNN model, clean set $\mathcal{X}_c$ and noisy set $\mathcal{X}_u$ are separated via the widely used low-loss criterion (Han et al., 2018b; Wei et al., 2020), where a dynamic memory rate $R(t) \in [0,1]$ is set for DNN to gradually distinguish $(1 - R(t))$ data with highest loss value as noisy samples while keeping other samples as clean set. Note that this simple separation criterion can not strictly eliminate noisy samples (see Fig. 1). Hence we choose to handle them respectively, where $\mathcal{X}_c$ update DNN parameters via sample credibility guided loss adjustment (3.1), and $\mathcal{X}_u$ is leveraged via another label learning scheme(3.2). A detailed algorithm pipeline can be found at Appendix A.1.

3.1 Sequential credibility modeling

Sequential credibility analysis. Previous works prefer to assess the data reliability purely based on its statistical property on a single point of time (e.g. the loss value in current epoch) during training process, i.e. they regard the credibility $w(x,y)$ of the i-th data sample (x,y) proportional to the joint distribution of its data-label pair,

$$w(x,y) \propto P(x,y). \tag{1}$$

However, as shown in Fig. 1, the training curve of normal and noisy samples usually yield different statistic information, where noisy samples usually have relatively larger loss values and poorer prediction consistency compared with clean ones, therefore the historical record of data training is also informative enough to help distinguish noisy and clean data.

This observation inspires us to estimate the data credibility in a sequential manner. To be specific, we define a sequence with length n as:

$$\mathbf{L}_t^n = [\mathbf{f}_t, \mathbf{f_{t-1}}, \cdots, \mathbf{f}_{t-n+1}], \quad \mathbf{f}_t = f\left(x; \theta_t\right). \tag{2}$$

Eq (2) illustrates a sliding window covering the feature snapshot of data from previous n epochs to current time point, where θ_t denotes the model parameters at the t-th epoch, and we model the data credibility with the likelihood and consistency of this historical sequences,

$$w(x, y) \propto C(\mathbf{L}_t^n, y) \log P\left(\mathbf{L}_t^n | \mathbf{f}_{t-n}, y\right). \tag{3}$$

The Eq (3) can be decoupled with two items, where $C(\mathbf{L}_t^n, y)$ measures the stability of training sequence given its label y, while $\log P\left(\mathbf{L}_t^n | \mathbf{f}_{t-n}, y\right)$ denotes log-likelihood of sequence generated from the $(t-n)$-th data observation of neural network training process. To estimate the sequential log-likelihood, we further assume that the observation in sequence $\mathbf{L}_t^n$ conforms to a certain markov property as:

$$\mathbf{f}_t \perp \mathbf{f}_i | (\mathbf{f}_{t-1}, y) \quad \forall \quad i < t-1. \tag{4}$$

The assumption in Eq (4) is reasonable since in most iterative learning algorithm like SGD, the data feature distribution is only decided by last observation and its label. With this assumption, we can further derive the likelihood as:

$$\begin{aligned}
\log P\left(\mathbf{L}_t^n | \mathbf{f}_{t-n}, y\right) &= \log P\left(\mathbf{f}_t | \mathbf{L}_{t-1}^n, y\right) + \log P\left(\mathbf{L}_{t-1}^{n-1} | \mathbf{f}_{t-n}, y\right) \\
&= \sum_{i=0}^{n-1} \log P\left(\mathbf{f}_{t-i} | \mathbf{L}_{t-i-1}^{n-i}, y\right) \\
&= \sum_{i=0}^{n-1} \log P\left(\mathbf{f}_{t-i} | \mathbf{f}_{t-i-1}, y\right).
\end{aligned} \tag{5}$$

With Eq (5), we can represent the sequential likelihood as the summation of conditional likelihood of data observation at each iteration within a sliding window of length n. In implementation, we can apply a normalized mixture model like GMM or BMM (Arazo et al., 2019; Li et al., 2020) as estimator to estimate the conditional probability $P\left(\mathbf{f}_{t-i} | \mathbf{f}_{t-i-1}, y\right)$ in Eq (5). Meanwhile, with the conditional probability estimation, the stability measurement $C(\mathbf{L}_t^n, y)$ is further designed as a modulator to suppress loss on training sequence with intense fluctuation,

$$C(\mathbf{L}_t^n, y) = 1 - \sqrt{\frac{1}{n} \sum_{i=0}^{n-1} \left(P\left(\mathbf{f}_{t-i} | \mathbf{f}_{t-i-1}, y\right) - \bar{P}\left(\mathbf{L}_t^n, y\right)\right)^2}. \tag{6}$$

$$\bar{P}\left(\mathbf{L}_t^n, y\right) = \frac{1}{n} \sum_{i=0}^{n-1} P\left(\mathbf{f}_{t-i} | \mathbf{f}_{t-i-1}, y\right). \tag{7}$$

Adaptively loss adjustment. The sequential likelihood $\bar{P}\left(\mathbf{L}_t^n\right)$ and stability measurement $C(\mathbf{L}_t^n, y)$ reflects how confident of the sample being clean. With the estimated credibility we reweight loss to update DNN as:

$$\theta_{t+1} = \theta_t - \eta \nabla \Big(\frac{1}{|\mathcal{X}_c|} \sum_{(x,y) \in \mathcal{X}_c} w(x, y) \mathcal{L}\big(f(x; \theta_t), y\big) \Big). \tag{8}$$

Where $\mathcal{L}$ is the objective function. $w(x, y)$ is the sample credibility and it modulates the contribution of each sample through gradient descending algorithm. Note that Eq. 8 is only applied on clean set $\mathcal{X}_c$, in this way, negative impact of hard noisy samples within $\mathcal{X}_c$ can be mitigated.

Objective function. Inspired by the design of symmetric cross entropy (SCE) function (Wang et al., 2019), a symmetric JS-divergence function with a co-regularization term is leveraged in *CREMA* as:

$$\mathcal{L} = D_{\text{JS}}(y || h(f_1(x; \theta))) + D_{\text{JS}}(y || h(f_2(x; \theta))) + D_{\text{JS}}\left(h(f_1(x; \theta)) || h(f_2(x; \theta))\right), \tag{9}$$

Where, $h(x)$ is the softmax probabilities produced by model. The reason we choose JS-divergence instead of cross entropy (CE) as loss function is that, CE tend to over-fit noisy samples as these samples cause relatively large gradient values during training. JS-divergence mitigates this problem via using predictions as supervising signals as well. Since for noisy samples, DNN predictions are usually more reliable that its label. Following previous work (Tanaka et al., 2018), a prior label distribution term and a negative entropy term are included to regularize training and further alleviate the over-fitting problem.

3.2 Selective label distribution learning

Following the divide-and-conquer idea, we attempt to leverage the separated noisy samples $\mathcal{X}_u$ as well. Some hard clean samples are blended with the separated noisy ones. Thus instead of discarding these wrongly labeled data as in sample detection methods (Han et al., 2018b; Wei et al., 2020), we resort to label correction approaches (Tanaka et al., 2018; Yi & Wu, 2019) to exploit them with gradually corrected labels and further boost performance.

Specifically, labels of $\mathcal{X}_u$ are treated as extra parameters, and updated through back-propagation to optimize a certain objective, this means both the network parameters and labels are updated simultaneously during training process, where original one-hot labels y will turn into a soft label distribution $\tilde{y} = h(y)$ after updating. Formally, $\tilde{y}$ is updated as $\tilde{y} \leftarrow \tilde{y} - \lambda(\partial \mathcal{L}_l / \partial \tilde{y})$. Where λ is learning rate and $\mathcal{L}_l$ is the objective to supervise the label correction process as $\mathcal{L}_l = D_{\text{JS}}(h(f_1(x;\theta)||\tilde{y}) + D_{\text{JS}}(h(f_2(x;\theta)||\tilde{y})$.

Empirical insight. In our experiments. we find that global label learning strategy (i.e., correcting labels of all training data) suffers from correction error in clean data. This can be observed from Fig. 3 (a), large gradient value will also be imposed on lots of correctly-labeled samples, consequently labels for these clean samples are unnecessarily updated. Compared with global label correction manners, we choose to only update the separated noisy samples $\mathcal{X}_u$ (mostly noisy). As shown in Fig. 3 (b), the proposed selective label correction strategy focuses more on learning noisy labels. The number of correctly-labeled samples with large gradient value is way less than a global correction scheme. Experiments in Sec 4.3 also quantitatively verify the effectiveness of the selective label learning strategy over global label learning manner.

4 Experiments

4.1 Datasets and Implementation Details

Datasets. To validate the effectiveness of the proposed method, we experimentally investigate on three synthetic noisy datasets, i.e., MNIST, CIFAR-10, CIFAR-100 (Krizhevsky et al., 2009) and two real-world label noise datasets, i.e., Clothing1M (Xiao et al., 2015a), and Animal10N (Song et al., 2019). MNIST consists of 70,000 images of size 28 × 28 for 10 classes, in which 60,000 images for training and the left 10,000 images for testing. Both CIFAR-10 and CIFAR100 contain 50,000 training images and 10,000 testing images of size $32 \times 32 \times 3$. Differently, the former has 10 classes, while CIFAR-100 has 100 classes. For the Clothing1M, it is a large-scale real-world noisy dataset which is collected from multiple online shopping websites. It contains 1 million training images and clean training subsets (47K for training, 14K for validation and 10K for test) with 14 classes. Noise rate for this dataset is around 38.5%. Animal-10N contains 55,000 human labeled online images for 10 confusing animals. It contains approximately 8% noisy samples. Following previous works (Song et al., 2019; Zhang et al., 2021), 50,000 images are exploited as a training set while the left for testing.

Implementation Details. For the three synthetic noisy datasets, we follow the setting in previous works (Han et al., 2018b; Yu et al., 2019; Wei et al., 2020), experiments with three kind of noise type are considered, i.e., symmetric noise (uniformly random), asymmetric noise and pairflip noise. Specifically, symmetric noise is generated by replacing labels in each class with labels of other classes uniformly. Asymmetric noise simulates fine-grained classification (for example, lynx and cat in Animal-10N (Song et al., 2019) with noisy labels, where labels are corrupted to a set of similar classes. Pairflip noise is generated by flipping each class to its adjacent class. Varying noise rates τ are conducted to fully evaluate the proposed method, where $\tau \in \{20\%, 50\%, 80\%\}$ for symmetric label noise , $\tau = 40\%$ for asymmetric noise and $\tau \in \{40\%, 45\%\}$ for pairflip label noise.

Table 1: Average test accuracy (%) on *MNIST* over the last ten epochs.

Noise rates τ	Standard	PENCIL	Co-teaching	Co-teaching+	JoCoR	CREMA (ours)
Symmetry-20%	79.94 ± 0.10	97.20 ± 0.53	97.40 ± 0.09	97.81 ± 0.03	97.98 ± 0.02	**98.40** ± 0.14
Symmetry-50%	52.92 ± 0.21	96.22 ± 0.13	92.47 ± 0.14	95.80 ± 0.09	96.35 ± 0.02	**98.07** ± 0.24
Symmetry-80%	23.95 ± 0.18	87.64 ± 0.25	82.04 ± 0.43	58.92 ± 0.37	85.51 ± 0.08	**92.02** ± 0.54
Asymmetry-40%	78.80 ± 0.09	94.39 ± 0.37	90.57 ± 0.04	93.28 ± 0.43	94.14 ± 0.12	**97.15** ± 0.26
Pairflip-40%	58.51 ± 0.29	94.06 ± 0.09	90.73 ± 0.22	89.91 ± 0.31	93.47 ± 0.10	**95.80** ± 0.51
Pairflip-45%	54.54 ± 0.30	90.73 ± 0.29	89.42 ± 0.22	85.81 ± 0.30	91.30 ± 0.25	**94.12** ± 0.58

Table 2: Average test accuracy (%) on *CIFAR-10* over the last ten epochs.

Noise rates τ	Standard	PENCIL	Co-teaching	Co-teaching+	JoCoR	CREMA (ours)
Symmetry-20%	68.67 ± 0.11	78.78 ± 0.15	82.56 ± 0.24	82.27 ± 0.21	85.73 ± 0.19	**86.32** ± 0.16
Symmetry-50%	42.31 ± 0.18	64.71 ± 0.27	72.97 ± 0.22	63.01 ± 0.33	79.53 ± 0.10	**81.63** ± 0.13
Symmetry-80%	15.94 ± 0.07	26.96 ± 0.37	24.03 ± 0.18	17.96 ± 0.06	27.30 ± 0.08	**29.66** ± 0.16
Asymmetric-40%	70.04 ± 0.08	70.06 ± 0.28	75.96 ± 0.15	72.21 ± 0.43	76.31 ± 0.21	**82.49** ± 0.13
Pairflip-40%	51.66 ± 0.11	75.26 ± 0.18	75.10 ± 0.23	57.59 ± 0.45	68.56 ± 0.16	**85.00** ± 0.13
Pairflip-45%	45.78 ± 0.13	71.18 ± 0.28	70.68 ± 0.23	49.60 ± 0.23	57.68 ± 0.21	**82.94** ± 0.12

Table 3: Average test accuracy (%) on *CIFAR-100* over the last ten epochs.

Noise rates τ	Standard	PENCIL	Co-teaching	Co-teaching+	JoCoR	CREMA (ours)
Symmetry-20%	34.72 ± 0.07	52.11 ± 0.21	50.48 ± 0.24	49.27 ± 0.03	53.41 ± 0.09	**57.21** ± 0.25
Symmetry-50%	16.86 ± 0.09	39.89 ± 0.30	38.24 ± 0.26	40.04 ± 0.70	43.37 ± 0.09	**43.95** ± 0.42
Symmetry-80%	4.60 ± 0.12	16.08 ± 0.15	11.78 ± 0.12	13.44 ± 0.37	12.33 ± 0.13	**17.10** ± 0.19
Asymmetric-40%	26.93 ± 0.10	32.81 ± 0.23	33.36 ± 0.28	33.62 ± 0.39	32.66 ± 0.13	**38.61** ± 0.25
Pairflip-40%	27.48 ± 0.12	33.83 ± 0.52	33.94 ± 0.18	33.80 ± 0.25	33.89 ± 0.12	**38.06** ± 0.34
Pairflip-45%	24.21 ± 0.11	29.01 ± 0.28	29.57 ± 0.15	26.93 ± 0.34	28.83 ± 0.10	**32.50** ± 0.29

For real-world noisy Clothing1M dataset, following Yi & Wu (2019); Zhang et al. (2021), we do not use the 50K clean data, and a randomly sampled pseudo-balanced subset includes about 260K images is leveraged as training data.

For the network structure, a 9-layer CNN with Leaky-ReLU activation function (Han et al., 2018b) is used for MNIST, CIFAR-10 and CIFAR-100, while ResNet-50 is adopted for Clothing1M and Animal-10N datasets. The batch size is set as 64 for all the datasets. For fair comparisons, we train our model for 200 epochs in total and choose the average test accuracy of last 10 epochs as the final result in three synthetic noisy datasets. Total training epochs for Clothing1M and Animal-10N are 80 and 150 respectively. Additionally, all the methods are implemented in PyTorch and run on NVIDIA Tesla V100 GPUs. Moreover, we use Adam optimizer for all the experiments and set the initial learning rate as 0.001, then it is degraded by a factor of 5 every 30 epochs for Clothing1M and 50 epochs for Animal-10N. The two classifiers in our methods are two networks with the same structure but different initialization parameters. Following Han et al. (2018b), $R(t)$ is linearly decreased along with training until reach a lower bound value σ, for Clothing1M and Animal-10N, we empirically set lower bound σ as 0.8 and 0.92 respectively.

4.2 Comparison with state-of-the-art methods

Results on synthetic noisy datasets. Table 1, Table 2, and Table 3 show the detailed results of the proposed *CREMA* and other methods in multiple synthetic noisy cases on three widely used datasets, i.e., MNIST, CIFAR-10, CIFAR-100. Specifically, four state-of-the-art LNL methods are chosen for comparison: PENCIL (Yi & Wu, 2019), Co-teaching (Han et al., 2018b), Co-teaching+ (Yu et al., 2019), JoCoR (Wei et al., 2020). Standard DNN training with cross entropy is also included as baseline. All the results of these methods are reproduced with their public code and suggested hyper-parameters for fair comparison. From these tables, we can observe that all these methods show better performance than Standard in the most natural Symmetry-20% case, which verifies

Table 4: Comparison with state-of-the-art methods in test accuracy on Clothing1M. “LA”, “LC” and “ND” denote “Loss Adjustment”, “Label Correction” and “Noisy sample Detection” respectively. Results for baselines are quoted from original papers.

Method	Category			Test Accuracy (%)
	LA	LC	ND	
Cross-Entropy				69.21
GCE (Zhang & Sabuncu, 2018)	✓			69.75
SCE (Wang et al., 2019)	✓			71.02
F-correction (Patrini et al., 2017)	✓			69.84
M-correction (Arazo et al., 2019)	✓			71.00
Masking (Han et al., 2018a)	✓			71.10
Joint-Optim (Tanaka et al., 2018)		✓		72.23
PENCIL (Yi & Wu, 2019)		✓		73.49
PLC (Zhang et al., 2021)		✓		74.02
Self-Learning (Han et al., 2019)		✓		74.45
Co-teaching (Han et al., 2018b)			✓	70.15
JoCoR (Wei et al., 2020)			✓	70.30
C2D (Zheltonozhskii et al., 2021)			✓	74.30
DivideMix (Li et al., 2020)			✓	**74.76**
CREMA (Ours)	✓	✓	✓	74.53

Table 5: Comparison with state-of-the-art methods in test accuracy on Animal-10N. short for category is the same as Table 4. Results for baselines approaches are quoted from Song et al. (2019) and Zhang et al. (2021).

Method	Category			Test Accuracy (%)
	LA	LC	ND	
Cross-Entropy				79.4
ActiveBias (Chang et al., 2017)	✓			80.5
PLC (Zhang et al., 2021)		✓		83.4
Co-teaching (Han et al., 2018b)			✓	80.2
SELFIE (Song et al., 2019)		✓	✓	81.8
CREMA (Ours)	✓	✓	✓	**84.2**

their robustness. Among them, JoCoR and PENCIL perform much better over other methods. However, when it comes to Pairflip-40% and Pairflip-45% cases, their performance drops significantly. On the contrary, the proposed *CREMA* can achieve consistent improvements over other methods on three benchmarks across various noise settings. In the Pairflip-40% and Pairflip-45% cases, the proposed method outperforms other baselines by a large margin. Specifically, *CREMA* can achieve 16.44% and 25.26% improvement in accuracy over JoCoR on CIFAR-10. When dealing with extremely noisy scenario, e.g. Symmetry-80%, *CREMA* can also perform generally better than other methods.The result demonstrates the superiority of the proposed method across various types and levels of label noise.

Results on real-world noisy datasets. Experiments on real-word noisy datasets Clothing1M (Xiao et al., 2015a), Animal-10N (Song et al., 2019) are also conducted to verify the effectiveness of the proposed method. The baseline methods are chosen from recently proposed LNL methods. Specifically, several loss adjustment methods, including GCE (Zhang & Sabuncu, 2018), SCE (Wang et al., 2019), F-correction (Patrini et al., 2017), M-correction (Arazo et al., 2019), Masking (Han et al., 2018a), ActiveBias (Chang et al., 2017), label correction methods, including Joint-Optim (Tanaka et al., 2018), PENCIL (Yi & Wu, 2019), Self-Learning (Han et al., 2019), PLC (Zhang et al., 2021), noisy sample detection approaches, including Co-teaching (Han et al., 2018b), JoCoR (Wei et al., 2020), C2D (Zheltonozhskii et al., 2021), DivideMix (Li et al., 2020) and a hybrid method SELFIE (Song et al., 2019) are compared with the proposed method. Table 4 and Table 5 show results on two real-world noisy datasets respectively. On large-scale Clothing1M dataset, *CREMA* outperforms all compared methods and achieves comparable performance with DivideMix. Note that DivideMix requires more training time, since it involves multiple rounds of forward computations within a single training iteration. While *CREMA* follows the standard DNN training procedure, and is similar to other co-training methods (Han et al., 2018b; Yu et al., 2019; Wei et al., 2020) in terms

Table 6: Ablation studies of each component within *CREMA* on Clothing1M dataset.

Method	Test Accuracy (%)
Baseline	72.81
+ Selective label update	73.25
+ Sequential likelihood	74.00
+ Stability measurement	**74.53**

Table 7: Investigations on different mixture models on Clothing1M dataset.

Estimator	Test Accuracy (%)
BMM	74.09
GMM	**74.53**

Table 8: Investigations on length of sequence n on Clothing1M dataset.

Length of sequence n	1	2	3	4	5	6
Test Accuracy (%)	73.25	73.99	**74.53**	74.40	74.27	73.96

of training time, since the time cost for sample credibility modeling is negligible compared with DNN update. State-of-the-art result is achieved by *CREMA* in Animal-10N as well. This indicates that the proposed *CREMA* can work well on high noise level (i.e., Clothing1M) and fine-grained (i.e., Animal-10N) real-world noisy datasets across various noise levels.

4.3 Ablation studies

• **Component Analysis.** *CREMA* contains several important components, including selective label learning strategy, sequential likelihood $\log P\left(\mathbf{L}_t^n|\mathbf{f}_{t-n}, y\right)$ and stability measurement $C(\mathbf{L}_t^n, y)$. To verify the effectiveness of each component, we conduct experiments on large-scale noisy dataset Clothing1M. The baseline method is built upon a simple co-teaching framework (Wei et al., 2020) combined with global label correction scheme (as in Yi & Wu (2019)), without the credibility guided loss adjustment strategy. The results are shown in Table 6, we can see that, conform to the observation on Fig. 3, the proposed selective label learning strategy achieves better result compared with the global correction counterpart. The sequential likelihood and stability measurement further boost the model performance with 0.75% and 0.53% accuracy gain, this indicate that the proposed sequential sample credibility modeling can effectively combat hard noisy samples mixed with clean ones. With all the three key components above, *CREMA* can achieve 74.53% test accuracy on Clothing1M.

• **Length of sequence n.** We also conduct experiments to investigate how the length of sequence n affects the performance. Table 8 shows results on Clothing1M with various values of n. It can be observed that increasing the length of sequence helps achieve higher accuracy at first but turn poor after hitting the peak value. Intuitively, no temporal information is provided when $n = 1$, *CREMA* can not utilize consistency metric, thus leading to a inferior result. When n is larger than 4, we also notice that performance degrades, this is probably due to unreliable model inside the very long sequence can do harm to sample credibility modeling and reversely hinder the final result.

• **Effect of different estimators.** Normalized mixture model plays the roles of estimating the conditional probability $P\left(\mathbf{f}_t|\mathbf{f}_{t-1}, y\right)$ in Eq (5). We compare two different estimators, Gaussian Mixture Model (GMM) (Permuter et al., 2006) and Beta Mixture Model (BMM) (Arazo et al., 2019) on Clothing1M. Table 7 shows the results. We can see that GMM obtain a relatively higher test accuracy, but BMM can also achieve good result (74.09%) as well. This indicate that the choice of probabilistic model is not sensitive to the final result.

5 Conclusion

In this paper, we propose a novel end-to-end robust learning method, called *CREMA*. Towards the problem that previous works lack the consideration of intrinsic difference among difficulties of noisy samples. *CREMA* follows the idea of divide-and-conquer that separate clean and noisy samples via estimating the credibility of each training sample. Two different branches are designed to handle the imperfectly separated samples respectively. For easily recognizable noisy samples, we apply a selective label correction scheme that avoiding erroneous label update on clean samples. For hard noisy samples that blended with clean ones, likelihood estimation of historical credibility sequence adaptively modulate the loss term of each sample during training. Extensive experiments conducted on synthetic and real-world noisy datasets demonstrate the superiority of the proposed method.

References

Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Unsupervised label noise modeling and loss correction. In *Proc. International Conference on Machine Learning (ICML)*, 2019. 3, 5, 8, 9

Devansh Arpit, Stanisław Jastrzebski, 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 *Proc. International Conference on Machine Learning (ICML)*, pp. 233–242, 2017. 1

David Berthelot, Nicholas Carlini, Ian J. Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 4

Haw-Shiuan Chang, Erik Learned-Miller, and Andrew McCallum. Active Bias: Training more accurate neural networks by emphasizing high variance samples. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1002–1012, 2017. 3, 4, 8

Mostafa Dehghani, Aliaksei Severyn, Sascha Rothe, and Jaap Kamps. Avoiding your teacher's mistakes: Training neural networks with controlled weak supervision. *arXiv preprint arXiv:1711.00313*, 2017a. 3

Mostafa Dehghani, Aliaksei Severyn, Sascha Rothe, and Jaap Kamps. Learning to learn from weak supervision by full supervision. In *Proc. Advances in Neural Information Processing Systems Workshop (NeurIPSW)*, 2017b. 3

Aritra Ghosh, Himanshu Kumar, and PS Sastry. Robust loss functions under label noise for deep neural networks. In *Proc. Association for the Advancement of Artificial Intelligence (AAAI)*, 2017. 3

Jacob Goldberger and Ehud Ben-Reuven. Training deep neural-networks using a noise adaptation layer. In *Proc. International Conference on Learning Representations (ICLR)*, 2017. 3

Bo Han, Jiangchao Yao, Gang Niu, Mingyuan Zhou, Ivor Tsang, Ya Zhang, and Masashi Sugiyama. Masking: A new perspective of noisy supervision. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5836–5846, 2018a. 3, 8

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. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pp. 8527–8537, 2018b. 1, 2, 3, 4, 6, 7, 8, 13

Jiangfan Han, Ping Luo, and Xiaogang Wang. Deep self-learning from noisy labels. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pp. 5138–5147, 2019. 3, 8

Dan Hendrycks, Mantas Mazeika, Duncan Wilson, and Kevin Gimpel. Using trusted data to train deep networks on labels corrupted by severe noise. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pp. 10456–10465, 2018. 1, 3

Michael E Houle. Local intrinsic dimensionality I: An extreme-value-theoretic foundation for similarity applications. In *Proc. International Conference on Similarity Search and Applications (SISAP)*, pp. 64–79, 2017. 3

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 *Proc. International Conference on Machine Learning (ICML)*, 2018. 1, 3, 4

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 6

Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. *arXiv preprint arXiv:2002.07394*, 2020. 4, 5, 8

Yuncheng Li, Jianchao Yang, Yale Song, Liangliang Cao, Jiebo Luo, and Li-Jia Li. Learning from noisy labels with distillation. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pp. 1910–1918, 2017. 3

Or Litany and Daniel Freedman. Soseleto: A unified approach to transfer learning and training with noisy labels. *arXiv preprint arXiv:1805.09622*, 2018. 3

Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 38(3):447–461, 2015. 3

Yueming Lyu and Ivor W Tsang. Curriculum loss: Robust learning and generalization against label corruption. In *Proc. International Conference on Learning Representations (ICLR)*, 2020. 3

Xingjun Ma, Yisen Wang, Michael E Houle, Shuo Zhou, Sarah M Erfani, Shu-Tao Xia, Sudanthi Wijewickrema, and James Bailey. Dimensionality-driven learning with noisy labels. In *Proc. International Conference on Machine Learning (ICML)*, 2018. 3

Eran Malach and Shai Shalev-Shwartz. Decoupling" when to update" from" how to update". In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pp. 960–970, 2017. 3

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. In *Proc. International Conference on Learning Representations (ICLR)*, 2020. 3

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 *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1944–1952, 2017. 1, 3, 8

H. Permuter, J. M. Francos, and I. Jermyn. A study of gaussian mixture models of color and texture features for image classification and segmentation. *Pattern Recognition (PR).*, 39:695–706, 2006. 9

Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. In *Proc. International Conference on Learning Representations (ICLR)*, 2015. 2, 3

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In *Proc. International Conference on Machine Learning (ICML)*, 2018. 3

Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-Weight-Net: Learning an explicit mapping for sample weighting. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pp. 1917–1928, 2019. 3

Hwanjun Song, Minseok Kim, and Jae-Gil Lee. SELFIE: Refurbishing unclean samples for robust deep learning. In *Proc. International Conference on Machine Learning (ICML)*, pp. 5907–5915, 2019. 3, 6, 8

Daiki Tanaka, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa. Joint optimization framework for learning with noisy labels. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5552–5560, 2018. 1, 3, 6, 8

Andreas Veit, Neil Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge Belongie. Learning from noisy large-scale datasets with minimal supervision. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 3

Ruxin Wang, Tongliang Liu, and Dacheng Tao. Multiclass learning with partially corrupted labels. *IEEE Transactions on Neural Networks and Learning Systems*, 29(6):2568–2580, 2017. 3

Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pp. 322–330, 2019. 3, 5, 8

Hongxin Wei, Lei Feng, Xiangyu Chen, and Bo An. Combating noisy labels by agreement: A joint training method with co-regularization. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13726–13735, 2020. 2, 3, 4, 6, 7, 8, 9

Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2691–2699, 2015a. 1, 6, 8

Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2691–2699, 2015b. 3

Quanming Yao, Hansi Yang, Bo Han, Gang Niu, and J. Kwok. Searching to exploit memorization effect in learning with noisy labels. In *Proc. International Conference on Machine Learning (ICML)*, 2020. 3

Kun Yi and Jianxin Wu. Probabilistic end-to-end noise correction for learning with noisy labels. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7017–7025, 2019. 1, 3, 6, 7, 8, 9

Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor W Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? In *Proc. International Conference on Machine Learning (ICML)*, 2019. 1, 3, 6, 7, 8

Chiyuan Zhang, Samy Bengio, Moritz Hardt, B. Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *Proc. International Conference on Learning Representations (ICLR)*, 2017. 1

Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *Proc. International Conference on Learning Representations (ICLR)*, 2018. 4

Yikai Zhang, Songzhu Zheng, Pengxiang Wu, Mayank Goswami, and Chao Chen. Learning with feature-dependent label noise: A progressive approach. In *Proc. International Conference on Learning Representations (ICLR)*, 2021. 3, 6, 7, 8

Zhilu Zhang and Mert Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pp. 8778–8788, 2018. 3, 8

Zizhao Zhang, Han Zhang, Sercan O Arik, Honglak Lee, and Tomas Pfister. Distilling effective supervision from severe label noise. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9294–9303, 2020a. 3

Zizhao Zhang, Han Zhang, Sercan O Arik, Honglak Lee, and Tomas Pfister. Distilling effective supervision from severe label noise. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9294–9303, 2020b. 3

Evgenii Zheltonozhskii, Chaim Baskin, Avi Mendelson, Alex M Bronstein, and Or Litany. Contrast to divide: Self-supervised pre-training for learning with noisy labels. *arXiv preprint arXiv:2103.13646*, 2021. 4, 8

Algorithm 1: *CREMA*. Line 5-9: sequential credibility modeling; Line 10-12: selective label update.

```
1  Input: network parameters $\theta^{(1)}$ and $\theta^{(2)}$, training dataset $\mathcal{D}$, dynamic memory rate $R(t)$, soft label
     distribution $\tilde{y}$, memory sequence $\mathbf{L}^n_{1,t}$ and $\mathbf{L}^n_{2,t}$.
2  while $t <$ MaxEpoch do
3      Fetch mini-batch $\mathcal{D}_n$ from $\mathcal{D}$;
4      Divide $\mathcal{D}_n$ into $\mathcal{X}_c$ and $\mathcal{X}_u$ based on $R(t)$;   // divide samples into clean and noisy
         set based on low-loss criterion
5      for $x_c \in \mathcal{X}_c$ do
6          Calculate $w(x_c, y_c)$ based on Eq (6) and Eq (7);    // sample credibility modeling
7
8          Update $\theta^{(1)}$ and $\theta^{(2)}$ based on Eq. (8);              // adaptive loss adjustment
9      end
10     for $x_u \in \mathcal{X}_u$ do
11         Update $\tilde{y}$, $\theta^{(1)}$, and $\theta^{(2)}$ through gradient descent;              // update soft label
             distribution and model parameters
12     end
13     Update $R(t)$;
14     Update $\mathbf{L}^n_{1,t}$ and $\mathbf{L}^n_{2,t}$;          // enqueue feature snapshot of current epoch
15 end
16 Output: $\theta^{(1)}$ and $\theta^{(2)}$.
```

A Appendix

A.1 Algorithm details

Algorithm 1 delineates the proposed *CREMA*. As illustrated in Sec. 3, *CREMA* is built on a divide-and-conquer framework. Firstly, clean set $\mathcal{X}_c$ and noisy set $\mathcal{X}_u$ are separated based on low-loss criterion (Han et al., 2018b). For $\mathcal{X}_c$, we compute the likelihood of historical credibility sequence, which helps to adaptively modulate the loss term of each training sample. As for $\mathcal{X}_u$, a selective label correction scheme is leveraged to update label distribution and model parameters simultaneously. After each training epoch, memory sequence $\mathbf{L}^n_{1,t}$ and $\mathbf{L}^n_{2,t}$ are updated with the feature snapshot of most current epoch.