How does Disagreement Help Generalization against Label Corruption?

Xingrui Yu[1], Bo Han[2,1], Jiangchao Yao[1], Gang Niu[2],
Ivor W. Tsang[1], and Masashi Sugiyama[2,3]

[1]Center for Artificial Intelligence, University of Technology Sydney, Australia
[2]Center for Advanced Intelligence Project, RIKEN, Japan
[3]Graduate School of Frontier Sciences, University of Tokyo, Japan
`{xingrui.yu,jiangchao.yao}@student.uts.edu.au`
`{bo.han,gang.niu}@riken.jp`
`ivor.tsang@uts.edu.au,sugi@k.u-tokyo.ac.jp`

Abstract. Learning with noisy labels is one of the hottest problems in weakly-supervised learning. Based on memorization effects of deep neural networks, training on small-loss instances becomes very promising for handling noisy labels. This fosters the state-of-the-art approach "Co-teaching" that cross-trains two deep neural networks using the small-loss trick. However, with the increase of epochs, two networks converge to a consensus and Co-teaching reduces to the self-training MentorNet. To tackle this issue, we propose a robust learning paradigm called Co-teaching+, which bridges the "Update by Disagreement" strategy with the original Co-teaching. First, two networks feed forward and predict all data, but keep prediction disagreement data only. Then, among such disagreement data, each network selects its small-loss data, but back propagates the small-loss data from its peer network and updates its own parameters. Empirical results on benchmark datasets demonstrate that Co-teaching+ is much superior to many state-of-the-art methods in the robustness of trained models. [1]

1 Introduction

In weakly-supervised learning, learning with noisy labels is one of the most challenging questions, since noisy labels are ubiquitous in our daily life, such as web queries [17], crowdsourcing [38], medical images [6], and financial analysis [1]. Essentially, noisy labels are systematically corrupted from ground-truth labels, which inevitably degenerates the accuracy of classifiers. Such degeneration becomes even more prominent for deep learning models (e.g., convolutional and recurrent neural networks), since these complex models can fully memorize noisy labels [41,2].

To handle noisy labels, classical approaches focus on either adding regularization [24] or estimating the label transition matrix [26]. Specifically, both explicit

[1] Preprint. Work in progress.

and implicit regularizations leverage the regularization bias to overcome the label noise issue. Nevertheless, they introduced a permanent regularization bias, and the learned classier barely reaches the optimal performance. Meanwhile, estimating the label transition matrix does not introduce the regularization bias, and the accuracy of classifiers can be improved by such accurate estimation. However, the label transition matrix is hard to be estimated, when the number of classes is large.

Recently, a promising way of handling noisy labels is to train on small-loss instances [11,30]. These works try to select small-loss instances, and then use them to update the network robustly. Among those works, the representative methods are MentorNet [11] and Co-teaching [9]. For example, MentorNet pre-trains an extra network, and then it uses the extra network for selecting clean instances to guide the training of the main network. When the clean validation data is not available, self-paced MentorNet has to use a predefined curriculum (e.g., small-loss instances). Nevertheless, the idea of self-paced MentorNet is similar to the self-training approach, and it inherits the same inferiority of accumulated error.

To solve the accumulated error issue in MentorNet, Co-teaching has been developed, which simultaneously trains two networks in a symmetric way [9]. First, in each mini-batch data, each network filters noisy (i.e., big-loss) samples based on the memorization effects. Then, it teaches the remaining small-loss samples to its peer network for updating the parameters, since the error from noisy labels can be reduced by peer networks mutually. From the initial training epoch, two networks having different learning abilities can filter different types of error. However, with the increase of training epochs, two networks will converge to a consensus gradually and Co-teaching reduces to the self-training MentorNet in function.

To address the consensus issue in Co-teaching, we should consider how to always keep two networks diverged within the training epochs, or how to slow down the speed that two networks will reach a consensus with the increase of epochs. Fortunately, we find that a simple strategy called “Update by Disagreement” [20] may help us to achieve the above target. This strategy conducts updates only on selected data, where there is a prediction disagreement between the two classifiers. By using “Update by Disagreement” strategy, we train two parallel networks on the clean data simultaneously. We can clearly envision that two networks trained by the “Update by Disagreement” strategy often keep diverged.

In this paper, we propose a robust learning paradigm called Co-teaching+ (Figure 1), which naturally bridges the “Update by Disagreement” strategy with Co-teaching. Co-teaching+ trains two deep neural networks similarly to the original Co-teaching, but it consists of the disagreement-update step (data update) and the cross-update step (parameters update). Initially, in the disagreement-update step, two networks feed forward and predict all data first, and only keep prediction disagreement data. Then, in the cross-update step, each network selects its small-loss data from such disagreement data, but back propagates the small-loss data from its peer network and updates its own parameters. Intu-

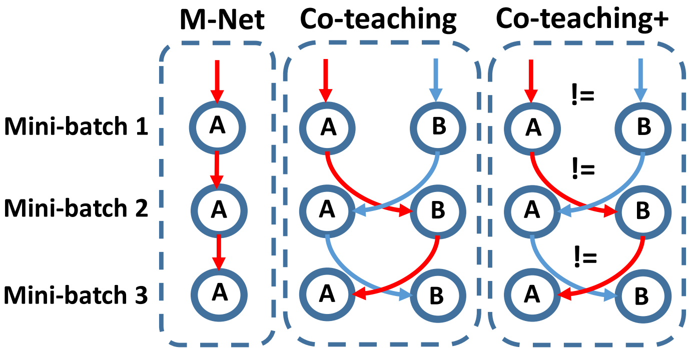


Fig. 1. Comparison of error flow among MentorNet (M-Net), Co-teaching and Co-teaching+. Assume that the error flow comes from the selection of training instances, and the error flow from network A or B is denoted by red arrows or blue arrows, respectively. **Left panel:** M-Net maintains only one network (A). **Middle panel:** Co-teaching maintains two networks (A & B) simultaneously. In each mini-batch data, each network selects its small-loss data to teach its peer network for the further training. **Right panel:** Co-teaching+ also maintains two networks (A & B). However, two networks feed forward and predict each mini-batch data first, and keep prediction disagreement data (!=) only. Based on such disagreement data, each network selects its small-loss data to teach its peer network for the further training.

itively, the idea of disagreement-update comes from Co-training [4], where two classifiers should keep diverged to achieve the better ensemble effects. The intuition of cross-update comes from culture evolving hypothesis [3], where a human brain can learn better if guided by the signals produced by other humans.

We conduct experiments on noisy versions of benchmark datasets, including *MNIST*, *CIFAR-10* and *NEWS*. Empirical results demonstrate that, from a low-level noisy case (i.e., 20% of noisy labels) to extremely noisy cases (i.e., 45% and 50% of noisy labels), the robustness of deep models trained by the Co-teaching+ approach is much superior to many state-of-the-art methods, including Co-teaching, MentorNet and F-correction [26].

Before delving into the details, we clearly emphasize our contribution in the following.

- We denote that "Update by Disagreement" (i.e., the Decoupling algorithm) itself *cannot* handle noisy labels, which has been empirically justified in Section 3.
- We realize that the "Update by Disagreement" strategy *can* keep two networks diverged, which significantly boosts the performance of Co-teaching.
- We summarize *three* key factors towards training robust deep networks with noisy labels: (1) using the small-loss trick; (2) cross-updating parameters of two networks; and (3) keeping two networks diverged.

The rest of this paper is organized as follows. In Section 2, we propose our robust learning paradigm Co-teaching+. Experimental results are discussed in

Section 3. In Section 4, we review the research progress in learning with noisy labels, and conclusions are given in Section 5.

2 Co-teaching+: Towards Training of Robust Deep Networks with Noisy Labels

Similar to Co-teaching, we also trains two deep neural networks. As in Figure 1, in each mini-batch data, each network conducts its own prediction, then selects instances for which there is a prediction disagreement between two networks. Based on such disagreement data, each network further selects its small-loss data, but back propagates the small-loss data selected by its peer network and updates itself parameters. We call such algorithm as Co-teaching+ (Algorithm 1), which consists of disagreement-update step and cross-update step. This brings the question as follows.

How does disagreement benefit Co-teaching? To answer this question, we should first understand the main drawback of Co-teaching. In the early stage of training, the divergence of two networks mainly comes from different (random) parameter initialization. Intuitively, this divergence between two networks pushes Co-teaching to become more robust than self-paced MentorNet, since two diverged networks have different abilities to filter different types of error. However, with the increase of training epochs, two networks will gradually converge to be close to each other. Therefore, Co-teaching degenerates to self-paced MentorNet gradually, and will not promote the learning ability to select clean data any more. To overcome this drawback, we need to keep the constant divergence between two networks or slow down the speed that two networks reach a consensus. This intuition comes from Co-training algorithm, where in semi-supervised learning [5], the better ensemble effects require to keep diverged more between two classifiers.

Fortunately, “Update by Disagreement” strategy [20] can help us to keep two networks diverged, since this strategy conducts algorithm updates only on selected data, where there is a prediction disagreement between the two classifiers. Therefore, within the whole training epochs, if two networks always select the disagreement data for further training, the divergence of two networks will be always maintained. Specifically, during the training procedure of Co-teaching, if we use “Update by Disagreement” strategy to keep two networks diverged, then we can prevent Co-teaching reducing to self-training MentorNet in function. This brings the new robust training paradigm Co-teaching+ (Algorithm 1).

Take “complementary peer learning” as an illustrative example for Co-teaching+. When students prepare for their exams, the peer learning will normally more boost their review efficiency than the solo learning. However, if two students are identically good at math but not good at literature, their review process in literature will have no any progress. Thus, the optimal peer should be complementary, which means that a student who is good at math should best review with another student who is good at literature. This point also explains why the diverged peer has more powerful learning ability than the identical peer.

Algorithm 1 Co-teaching+. Step 4: disagreement-update; Step 5-8: cross-update.

```
1: Input $w^{(1)}$ and $w^{(2)}$, training set $\mathcal{D}$, batch size $B$, learning rate $\eta$, estimated noise rate $\tau$, epoch $E_k$ and $E_{\max}$;
for $e = 1, 2, \dots, E_{\max}$ do
    2: Shuffle $\mathcal{D}$ into $\frac{|\mathcal{D}|}{B}$ mini-batches;    //noisy dataset
    for $n = 1, \dots, \frac{|\mathcal{D}|}{B}$ do
        3: Fetch $n$-th mini-batch $\bar{\mathcal{D}}$ from $\mathcal{D}$;
        4: Select prediction disagreement $\bar{\mathcal{D}}^{'}$ by Eq. (1);
        5: Get $\bar{\mathcal{D}}^{'(1)} = \arg\min_{\mathcal{D}':|\mathcal{D}'|\geq\lambda(e)|\bar{\mathcal{D}}^{'}|} \ell(\mathcal{D}'; w^{(1)})$;    //sample $\lambda(e)\%$ small-loss instances
        6: Get $\bar{\mathcal{D}}^{'(2)} = \arg\min_{\mathcal{D}':|\mathcal{D}'|\geq\lambda(e)|\bar{\mathcal{D}}^{'}|} \ell(\mathcal{D}'; w^{(2)})$;    //sample $\lambda(e)\%$ small-loss instances
        7: Update $w^{(1)} = w^{(1)} - \eta\nabla\ell(\bar{\mathcal{D}}^{'(2)}; w^{(1)})$;    //update $w^{(1)}$ by $\bar{\mathcal{D}}^{'(2)}$;
        8: Update $w^{(2)} = w^{(2)} - \eta\nabla\ell(\bar{\mathcal{D}}^{'(1)}; w^{(2)})$;    //update $w^{(2)}$ by $\bar{\mathcal{D}}^{'(1)}$;
    end
    9: Update $\lambda(e) = 1 - \min\{\frac{e}{E_k}\tau, (1 + \frac{e-E_k}{E_{\max}-E_k})\tau\}$;
end
10: Output $w^{(1)}$ and $w^{(2)}$.
```

Algorithm description. Algorithm 1 consists of the disagreement-update step (step 4) and the cross-update step (step 5-8), where we train two deep neural networks in a mini-batch manner.

In step 4, two networks feed forward and predict the same mini-bach of data $\bar{\mathcal{D}}$=$\{(x_1, y_1), (x_2, y_2), \cdots, (x_B, y_B)\}$ first, where the batch size is B. Then, they keep prediction disagreement data $\bar{\mathcal{D}}^{'}$ (Eq. (1)) according to their predictions $\{\bar{y}_1^{(1)}, \bar{y}_2^{(1)}, \dots, \bar{y}_B^{(1)}\}$ (prediction by $w^{(1)}$) and $\{\bar{y}_1^{(2)}, \bar{y}_2^{(2)}, \dots, \bar{y}_B^{(2)}\}$ (prediction by $w^{(2)}$):

$$\bar{\mathcal{D}}^{'} = \{(x_i, y_i) : \bar{y}_i^{(1)} \neq \bar{y}_i^{(2)}\}, \tag{1}$$

where $i \in \{1, \dots, B\}$. The intuition of this step comes from Co-training, where two classifiers should keep diverged to achieve the better ensemble effects.

In step 5-8, from the disagreement data $\bar{\mathcal{D}}^{'}$, each network $w^{(1)}$ (resp. $w^{(2)}$) selects its own small-loss data $\bar{\mathcal{D}}^{'(1)}$ (resp. $\bar{\mathcal{D}}^{'(2)}$), but back propagates the small-loss data $\bar{\mathcal{D}}^{'(1)}$ (resp. $\bar{\mathcal{D}}^{'(2)}$) to its peer network $w^{(2)}$ (resp. $w^{(1)}$) and updates parameters. The intuition of step 5-8 comes from the aforementioned culture evolving hypothesis [3], where a human brain can learn better if guided by the signals produced by other humans.

In step 9, we update $\lambda(e)$, which controls how many small-loss data should be selected in each training epoch. Due to the memorization effects, deep networks will fit clean data first and then gradually over-fit noisy data.

Thus, at the beginning of training, we keep more small-loss data (with a large $\lambda(e)$) in each mini-batch, which is equivalent to dropping less data. Since deep networks will fit clean data first, noisy data do not matter at the initial training epochs. With the increase of epochs, we keep less small-loss data (with a small

Table 1. Comparison of state-of-the-art and related techniques with our Co-teaching+ approach. In the first column, "small loss": regarding small-loss samples as "clean" samples; "deep networks": leveraging the memorization effects of deep neural networks; "double classifiers": training two classifiers simultaneously; "cross update": updating parameters in a cross manner instead of a parallel manner; "divergence": keeping two classifiers diverged during the whole training epochs.

	MentorNet	Co-training	Co-teaching	Decoupling	Co-teaching+
small loss	✓	✗	✓	✗	✓
neural networks	✓	✗	✓	✗	✓
double classifiers	✗	✓	✓	✓	✓
cross update	✗	✓	✓	✗	✓
divergence	✗	✓	✗	✓	✓

$\lambda(e)$) in each mini-batch. As deep networks will over-fit noisy data gradually, we should drop more data. The gradual decrease of $\lambda(e)$ prevents deep networks over-fitting noisy data to some degree.

Similar to Co-teaching, we decrease $\lambda(e)$ quickly at the first E_k epochs to stop networks over-fitting to the noisy data, namely $\lambda(e) = 1 - \frac{e}{E_k}\tau$. However, after E_k epochs, Co-teaching keeps a constant $\lambda(e)$, where $\lambda(e) = 1 - \tau$; while Co-teaching+ further decreases $\lambda(e)$ slowly, where $\lambda(e) = 1 - (1 + \frac{e - E_k}{E_{\max} - E_k})\tau$. To explain this difference, we take a working example here.

Assume that the estimated noise rate τ is 30%. It means that, after E_k epochs, Co-teaching will constantly fetch 70% small-loss data in each mini-batch as "clean" data. However, the τ estimation tends to be inaccurate in practice. Therefore, given the estimated τ, we should fetch less data, e.g., 60% small-loss data, to keep remained data more clean. This explains that Co-teaching+ further decreases $\lambda(e)$ slowly after E_k epochs.

Relations to other approaches. We compare our Co-teaching+ with related approaches in Table 1. We try to find the connections among them, and pinpoint the key factors that can handle noisy labels. First, self-paced MentorNet [11] employs the small-loss trick to handle noisy labels. However, this idea is similar to the self-training approach, and it inherits the same inferiority of accumulated error caused by the sample-selection bias. Inspired by Co-training [4] that trains double classifiers and cross updates parameters, Co-teaching [9] has been developed to cross train two deep networks, which addresses the accumulated error issue in MentorNet. Note that, Co-training does not exploit the memorization in deep neural networks, while Co-teaching does (i.e., leveraging small-loss trick).

However, with the increase of training epochs, two networks trained by Co-teaching will converge to a consensus, and Co-teaching will reduce to the self-training MentorNet. This brings us to think how to address the consensus issue in Co-teaching. Although Decoupling algorithm [20] (i.e., "Update by Disagreement") itself *cannot* combat with noisy labels effectively, which has been empirically justified in Section 3, we clearly realize that "Update by Disagreement" strategy can always keep two networks diverged. Such divergence effects can boost the performance of Co-teaching and bring us Co-teaching+, since the better ensemble effects require to keep diverged more between two classifiers due to Co-training.

To sum up, there are three key factors that can contribute to effectively handle noisy labels (first column of Table 1). First, we should leverage the memorization effects of deep networks (i.e., the small-loss trick). Second, we should train two deep networks simultaneously, and cross update their parameters. Last but not least, we should keep two deep networks diverged during the whole training epochs.

3 Experiments

3.1 Experimental setup

Datasets. We verify the effectiveness of our approach on three benchmark datasets (Table 2), including two vision datasets (i.e., *MNIST*, *CIFAR-10*) and one text dataset (i.e., *NEWS*). These data sets are popularly used for evaluation of noisy labels in the literature [29,7,12].

Table 2. Summary of data sets used in the experiments.

	# of training	# of testing	size of image/text
MNIST	60,000	10,000	28×28
CIFAR-10	50,000	10,000	32×32
NEWS	11,314	7,532	300-D

Since all datasets are clean, following [29,26], we need to corrupt these datasets manually by the label transition matrix Q, where $Q_{ij} = \Pr(\tilde{y} = j|y = i)$ given that noisy $\tilde{y}$ is flipped from clean y. Assume that the matrix Q has two representative structures:

(1) Symmetry flipping [34]; (2) Pair flipping [9]: a simulation of fine-grained classification with noisy labels, where labelers may make mistakes only within very similar classes.

Baselines. We compare the Co-teaching+ (Algorithm 1) with the following state-of-art approaches, and implement all methods with default parameters by PyTorch, and conduct all the experiments on a NVIDIA K80 GPU.

(i). MentorNet [11]. An extra teacher network is pre-trained and then used to filter out noisy instances for its student network to learn robustly under noisy labels. Then, student network is used for classification. We used self-paced MentorNet in this paper;

(ii). Co-teaching [9], which trains two networks simultaneously and cross-updates parameters of peer networks. This method can deal with a large number of classes and is more robust to extremely noisy labels;

Table 3. CNN and MLP models used in our experiments on *MNIST*, *CIFAR10*, and *NEWS*. The slopes of all LReLU functions in the networks are set to 0.001.

CNN on *MNIST*	CNN on *CIFAR10*	MLP on *NEWS*
28×28 Gray Image	32×32 RGB Image	300-D Embedding
3×3 conv, 128 LReLU 3×3 conv, 128 LReLU 3×3 conv, 128 LReLU		dense 300→300, Softsign
2×2 max-pool, stride 2 dropout, $p = 0.25$		
3×3 conv, 256 LReLU 3×3 conv, 256 LReLU 3×3 conv, 256 LReLU		
2×2 max-pool, stride 2 dropout, $p = 0.25$		dense 300→300
3×3 conv, 512 LReLU 3×3 conv, 256 LReLU 3×3 conv, 128 LReLU		
avg-pool		
dense 128→10	dense 128→10	dense 300→7

(iii). Decoupling [20], which updates the parameters only using the instances which have different prediction from two classifiers.

(iv). F-correction [26], which corrects the prediction by the label transition matrix. As suggested by the authors, we first train a standard network to estimate the transition matrix Q.

(v). As a simple baseline, we compare Co-teaching+ with the standard deep network that directly trains on noisy datasets (abbreviated as Standard).

Network structure. For *MNIST* and *CIFAR-10*, CNN is used with Leaky-ReLU (LReLU) active function [19]. Namely, the 9-layer CNN architecture in our paper follows "Temporal Ensembling" [13] and "Virtual Adversarial Training" [24]. For *NEWS*, we borrowed the pre-trained word embeddings from GloVe [27], and 3-layer MLP is used with Softsign active function. The network structure here is standard test bed for weakly-supervised learning, and the details are in Table 3.

Optimizer. Adam optimizer (momentum=0.9) is with an initial learning rate of 0.001, and the batch size is set to 128 and we run 200 epochs. Besides, dropout and batch-normalization are also used. As deep networks are highly nonconvex, even with the same network and optimization method, different initializations can lead to different local optimal. Thus, following [20], we also take two networks with the same architecture but different initializations as two classifiers.

Initialization. We assume the noise rate τ is known and set the ratio of small-loss samples $\lambda(e)$ as follows.

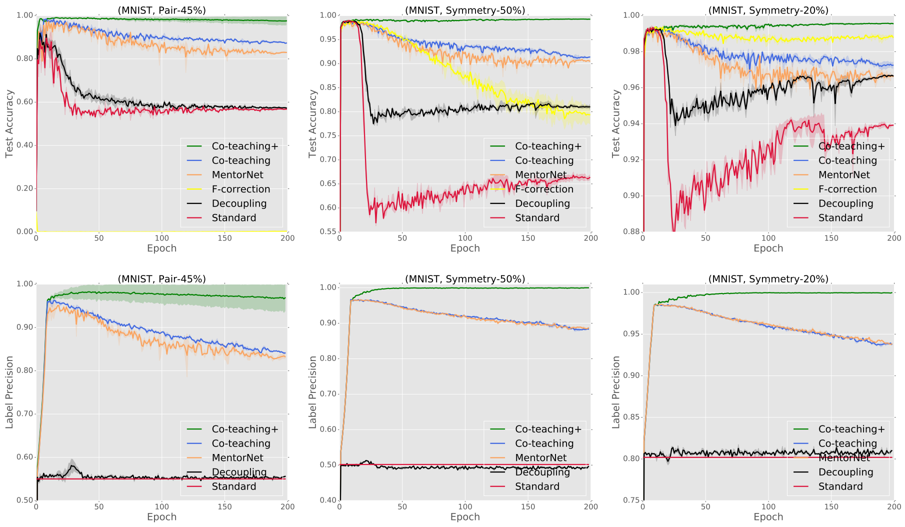
$$\lambda(e) = 1 - \min\{\frac{e}{E_k}\tau, (1 + \frac{e - E_k}{E_{\max} - E_k})\tau\}, \tag{2}$$

where $E_k = 10$ and $E_{\max} = 200$.

If τ is not known in advanced, τ can be inferred using validation sets [16,40]. Note that $\lambda(e)$ only depends on the memorization effect of deep networks but not any specific datasets.

Measurement. As for performance measurements, first, we use the test accuracy, i.e., *test Accuracy = (# of correct predictions) / (# of test dataset)*. Besides, we also use the label precision in each mini-batch, i.e., *label Precision = (# of clean labels) / (# of all selected labels)*. Specifically, we sample $\lambda(e)$ of small-loss instances in each mini-batch, and then calculate the ratio of clean labels in the small-loss instances. Intuitively, higher label precision means less noisy instances in the mini-batch after sample selection, and the algorithm with higher label precision is also more robust to the label noise.

3.2 Comparison with the State-of-the-Arts



(a) Pair-45%. (b) Symmetry-50%. (c) Symmetry-20%.

Fig. 2. Results on *MNIST* dataset. Top: test accuracy vs. number of epochs; bottom: label precision vs. number of epochs.

Results on MNIST. Figure 2 shows test accuracy and label precision vs. number of epochs on *MNIST*. In the top of Figure 2, we show test accuracy vs. number of epochs. In all three plots, we can clearly see the memorization effects of deep networks. For example, test accuracy of Standard first reaches a very high level since deep network will first fit clean labels. Over the increase of epochs, deep network will over-fit noisy labels gradually, which decreases its test accuracy accordingly. Thus, a robust training method should alleviate or even stop the decreasing trend in test accuracy.

In the easiest Symmetry-20% case, all new approaches work better than Standard obviously, which demonstrates their robustness. Co-teaching+ and F-correction work significantly better than Co-teaching, MentorNet and Decoupling. However, F-correction cannot combat with the other two harder cases, i.e., Pair-45% and Symmetry-50%. Especially in the hardest Pair-45% case, F-correction can learn nothing at all, which greatly restricts its practical usage in the wild. Besides, in two such cases, Co-teaching+ achieves higher accuracy than Co-teaching and MentorNet.

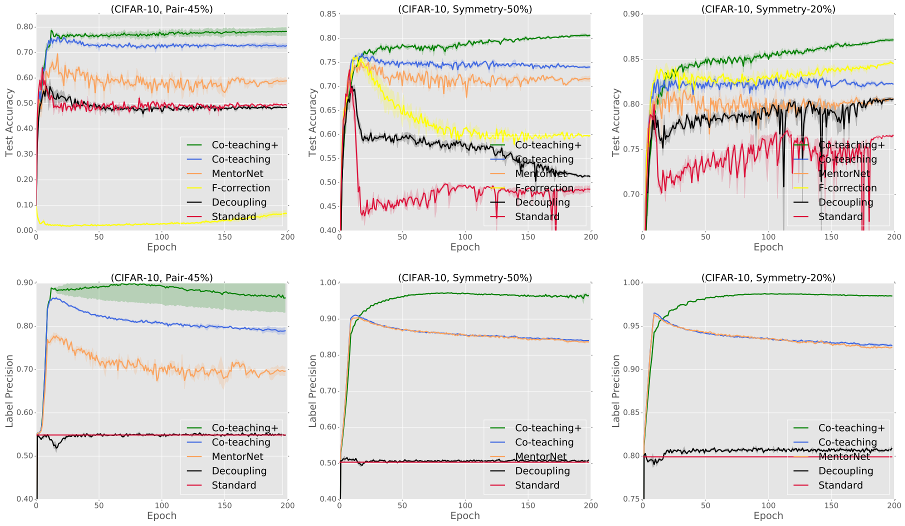
To explain such good performance, we plot label precision vs. number of epochs in the bottom of Figure 2. In principle, the higher label precision means the less noisy instances after sample selection, which naturally leads to the higher test accuracy during training. First, we can see both Standard and Decoupling do not act to pick up clean instances, which of course do not combat with noisy label at all. The reason is that both methods do not utilize the memorization effects during training. Then, we can clearly see that Co-teaching+, Co-teaching and MentorNet can successfully pick clean instances out.

However, our Co-teaching+ always achieve the highest label precision cross two easier cases to the hardest case, and this shows our new approach is better at finding clean instances. Therefore, the test accuracy of our Co-teaching+ is higher than others obviously. The other interesting point is that even when the label precisions of Co-teaching and MentorNet tie together in two easiest cases, i.e., Symmetry-20% and Symmetry-50%, the test accuracy of Co-teaching is still higher than that of MentorNet. This reveals that cross-training two networks has some regularization effects, which also helps the performance of Co-teaching+.

Results on *CIFAR-10*. Figure 3 shows test accuracy and label precision vs. number of epochs on *CIFAR-10*. In the top of Figure 3, we show test accuracy vs. number of epochs. Similarly, we can clearly see the memorization effects of deep networks, namely test accuracy of Standard first reaches a very high level then decreases gradually. In the easiest Symmetry-20% case, all new approaches work much better than Standard, which demonstrates their robustness. Co-teaching+ and F-correction work significantly better than Co-teaching, MentorNet and Decoupling.

However, F-correction cannot combat with two harder cases easily, i.e., Pair-45% and Symmetry-50%. In the Symmetry-50% case, F-correction works better than Standard and Decoupling, but much worse than the other three approaches. In the hardest Pair-45% case, F-correction almost learns nothing. In such two harder cases, our Co-teaching+ consistently achieves higher accuracy than Co-teaching and MentorNet. An interesting phenomenon is, in two easiest cases, Co-teaching+ not only fully stop the decreasing trend in test accuracy, but also performs better and better with the increase of epochs.

To explain such good performance, we plot label precision vs. number of epochs in the bottom of Figure 3. Similarly, the higher label precision leads to the higher test accuracy during training. Compared to Standard and Decoupling, another three approaches can successfully pick clean instances out. However, our Co-teaching+ always achieve the highest label precision cross three cases, which demonstrates that our new approach is better at finding clean instances. Thus, the test accuracy of our Co-teaching+ is higher than others. In two easiest cases, the label precision of Co-teaching+ increases fast to a high plateau and then keep constant. This may explain why the test accuracy of Co-teaching performs better and better with the increase of epochs.



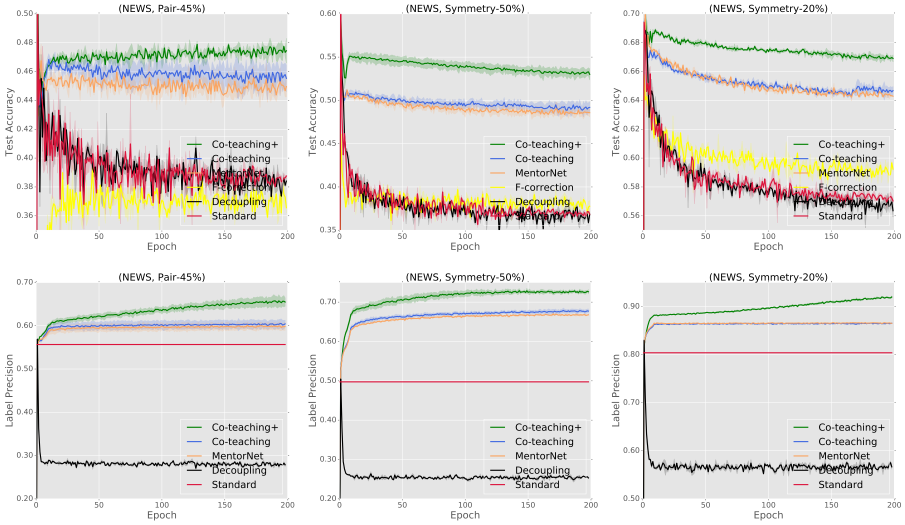
(a) Pair-45%. (b) Symmetry-50%. (c) Symmetry-20%.

Fig. 3. Results on *CIFAR-10* dataset. Top: test accuracy vs. number of epochs; bottom: label precision vs. number of epochs.

Results on *NEWS*. To verify our Co-teaching+ comprehensively, we conduct experiments not only on benchmark vision datasets, but also on benchmark text dataset *NEWS*. Figure 4 shows test accuracy and label precision vs. number of epochs on *NEWS*. In the top of Figure 4, we first show test accuracy vs. number of epochs.

Similar to results on vision datasets, we can still see the memorization effects of deep networks in all three plots, i.e., test accuracy of Standard first reaches a very high level and then gradually decreases. However, Co-teaching+ overcomes such memorization issue, and works much better than others across three cases. It is noted that, F-correction cannot combat with all three cases, even in the easiest Symmetry-20% case, not to mention harder cases.

To explain such good performance, we plot label precision vs. number of epochs in the bottom of Figure 4. Similarly, the higher label precision leads to the higher test accuracy during training. Compared to Standard and Decoupling, another three approaches can successfully pick clean instances out. However, in all three cases, Co-teaching+ always achieve the highest label precision, which demonstrates that our new approach is better at finding clean instances than Co-teaching and MentorNet. Thus, the test accuracy of our Co-teaching+ is higher than others.



(a) Pair-45%.
(b) Symmetry-50%.
(c) Symmetry-20%.

Fig. 4. Results on *NEWS* dataset. Top: test accuracy vs. number of epochs; bottom: label precision vs. number of epochs.

Reflection of results. Different algorithm designs lead to different results. To sum up, self-paced MentorNet is concluded as training single deep network using the small-loss trick. Co-teaching moves further step, which is viewed as cross-training double deep networks using the small-loss trick. Based on Co-teaching, Co-teaching+ is regarded as cross-training double *diverged* deep networks using the small-loss trick. Thus, keeping two deep networks diverged is one of the key ingredients to train robust deep networks. This point has been empirically verified by the result difference between Co-teaching and Co-teaching+.

4 Related literature

Statistical learning methods. Statistical learning contributed a lot to the problem of noisy labels, especially in theoretical aspects. Statistical learning approaches can be categorized into three strands: surrogate loss, noise rate estimation and probabilistic modeling. For example, in the surrogate losses category, [25] proposed an unbiased estimator to provide the noise corrected loss approach. [21]

presented a robust non-convex loss, which is the special case in a family of robust losses. In the noise rate estimation category, both [23] and [16] proposed a class-probability estimator using order statistics on the range of scores. [32] presented the same estimator using the slope of the ROC curve. In the probabilistic modeling category, [28] proposed a two-coin model to handle noisy labels from multiple annotators. [39] extended this two-coin model by setting the dynamic flipping probability associated with instances.

Deep learning approaches. Deep learning approaches are prevalent to handle noisy labels [42]. [15] proposed a unified framework to distill the knowledge from clean labels and knowledge graph, which can be exploited to learn a better model from noisy labels. [35] trained a label cleaning network by a small set of clean labels, and used this network to reduce the noise in large-scale noisy labels. [31] added a crowd layer after the output layer for noisy labels from multiple annotators. [33] presented a joint optimization framework to learn parameters and estimate true labels simultaneously. [30] leveraged an additional validation set to adaptively assign weights to training examples.

Similarly, based on a small set of trusted data with clean labels, [10] proposed a loss correction approach to mitigate the effects of label noise on deep neural network classifiers. [18] developed a new dimensionality-driven learning strategy, which monitors the dimensionality of deep representation subspaces during training and adapts the loss function accordingly. [37] proposed an iterative learning framework for training CNNs on datasets with open-set noisy labels. [8] proposed a human-assisted approach that conveys human cognition of invalid class transitions, and derived a structure-aware deep probabilistic model incorporating a speculated structure prior. [14] proposed a novel inference method to obtain a robust decision boundary under any softmax neural classifier pre-trained on noisy datasets. Their idea is to induce a generative classifier on top of hidden feature spaces of the discriminative deep model.

5 Conclusion

This paper presents a robust learning paradigm called Co-teaching+, which trains deep neural networks robustly under noisy supervision. Our key idea is to maintain two networks simultaneously that find the prediction disagreement data. Among such disagreement data, our method cross-trains on data screened by the “small loss” criteria. We conduct experiments to demonstrate that, our proposed Co-teaching+ can train deep models robustly with the extremely noisy supervision beyond Co-teaching and MentorNet. More importantly, we summarize three key points towards training robust deep networks with noisy labels: (1) using small-loss trick based on memorization effects of deep networks; (2) cross-updating parameters of two networks; and (3) keeping two deep networks diverged during the whole training epochs. In future, we will investigate the theory of Co-teaching+ from the view of disagreement-based algorithms [36].

Acknowledgments. MS was supported by JST CREST JPMJCR18A2. IWT was supported by ARC FT130100746, DP180100106 and LP150100671. Bo Han would like to thank the financial support from Center for Advanced Intelligence Project, RIKEN.

References

1. Aït-Sahalia, Y. and Fan, J. and Xiu, D.: High-frequency covariance estimates with noisy and asynchronous financial data. JASA. (2010)
2. Arpit, D. and Jastrzebski, S. and Ballas, N. and Krueger, D. and Bengio, E. and Kanwal, M.S. and Maharaj, T. and Fischer, A. and Courville, A. and Bengio, Y.: A closer look at memorization in deep networks. In: ICML. (2017)
3. Bengio, Y.: Evolving culture versus local minima. GAM. (2014)
4. Blum, A. and Mitchell, T.: Combining labeled and unlabeled data with co-training. COLT. (1998)
5. Chapelle, O. and Scholkopf, B. and Zien, A.: Semi-supervised learning. IEEE TNN. (2009)
6. Dgani, Y. and Greenspan, H. and Goldberger, J.: Training a neural network based on unreliable human annotation of medical images. In: ISBI. (2018)
7. Goldberger, J. and Ben-Reuven, E.: Training deep neural-networks using a noise adaptation layer. In: ICLR. (2017)
8. Han, B. and Yao, J. and Niu, G. and Zhou, M. and Tsang, I. and Zhang, Y. and Sugiyama, M.: Learning with noisy labels. In: NeurIPS. (2018)
9. Han, B. and Yao, J. and Niu, G. and Zhou, M. and Tsang, I. and Zhang, Y. and Sugiyama, M.: Co-teaching: Robust training of deep neural networks with extremely noisy labels. In: NeurIPS. (2018)
10. Hendrycks, D. and Mazeika, M. and Wilson, D. and Gimpel, K.: Using trusted data to train deep networks on labels corrupted by severe noise. In: NeurIPS. (2018)
11. Jiang, L. and Zhou, Z. and Leung, T. and Li, L. and Li, F.: MentorNet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In: ICML. (2018)
12. Kiryo, R. and Niu, G. and du Plessis, M. C and Sugiyama, M.: Positive-unlabeled learning with non-negative risk estimator. In: NeurIPS. (2017)
13. Laine, S. and Aila, T.: Temporal ensembling for semi-supervised learning. In: ICLR. (2017)
14. Lee, K. and Yun, S. and Lee, K. and Lee, H. and Li, B. and Shin, J.: Robust determinantal generative classifier for noisy labels and adversarial attacks. In: NeurIPS Workshop. (2018)
15. Li, Y. and Yang, J. and Song, Y. and Cao, L. and Luo, J. and Li, J.: Learning from noisy labels with distillation. In: ICCV. (2017)
16. Liu, T. and Tao, D.: Classification with noisy labels by importance reweighting. IEEE TPAMI. (2016)
17. Liu, W. and Jiang, Y. and Luo, J. and Chang, S.: Noise resistant graph ranking for improved web image search. In: CVPR. (2011)
18. Ma, X. and Wang, Y. and Houle, M. and Zhou, S. and Erfani, S. and Xia, S. and Wijewickrema, S. and Bailey, J.: Dimensionality-driven learning with noisy labels. In: ICML. (2018)
19. Maas, A. and Hannun, A. and Ng, A.: Rectifier nonlinearities improve neural network acoustic models. In: ICML. (2013)

20. Malach, E. and Shalev-Shwartz, S.: Decoupling" when to update" from" how to update". In: NeurIPS. (2017)
21. Masnadi-Shirazi, H. and Vasconcelos, N.: On the design of loss functions for classification: theory, robustness to outliers, and savageboost. In: NeurIPS. (2009)
22. Menon, A. and Van Rooyen, B. and Ong, C. and Williamson, B.: Learning from corrupted binary labels via class-probability estimation. In: ICML. (2015)
23. Menon, A. and Van Rooyen, B. and Ong, C. and Williamson, B.: Learning from corrupted binary labels via class-probability estimation. In: ICML. (2015)
24. Miyato, T. and Dai, A. and Goodfellow, I.: Virtual adversarial training for semi-supervised text classification. In: ICLR. (2016)
25. Natarajan, N. and Dhillon, I. and Ravikumar, P. and Tewari, A.: Learning with noisy labels. In: NeurIPS. (2013)
26. Patrini, G. and Rozza, A. and Menon, A. and Nock, R. and Qu, L.: Making deep neural networks robust to label noise: a loss correction approach. In: CVPR. (2017)
27. Pennington, J. and Socher, R. and Manning, C.: Glove: Global vectors for word representation. In: EMNLP. (2014)
28. Raykar, V. and Yu, S. and Zhao, L. and Valadez, G. and Florin, C. and Bogoni, L. and Moy, L.: Learning from crowds. JMLR. (2010)
29. Reed, S. and Lee, H. and Anguelov, D. and Szegedy, C. and Erhan, D. and Rabinovich, A.: Training deep neural networks on noisy labels with bootstrapping. In: ICLR workshop. (2015)
30. Ren, M. and Zeng, W. and Yang, B. and Urtasun, R.: Learning to reweight examples for robust deep learning. In: ICML. (2018)
31. Rodrigues, F. and Pereira, F.: Deep learning from crowds. In: AAAI. (2018)
32. Sanderson, T. and Scott, C.: Class proportion estimation with application to multiclass anomaly rejection. In: AISTATS. (2014)
33. Tanaka, D. and Ikami, D. and Yamasaki, T. and Aizawa, K.: Joint optimization framework for learning with noisy labels. In: CVPR. (2018)
34. van Rooyen, B. and Menon, A. and Williamson, R.C.: Learning with symmetric label noise: The importance of being unhinged. In: NeurIPS. (2015)
35. Veit, A. and Alldrin, N. and Chechik, G. and Krasin, I. and Gupta, A. and Belongie, S.: Learning from noisy large-scale datasets with minimal supervision. In: CVPR. (2017)
36. Wang, Wei and Zhou, Zhi-Hua: Theoretical foundation of co-training and disagreement-based algorithms. arXiv preprint arXiv:1708.04403. (2017)
37. Wang, Y. and Liu, W. and Ma, X. and Bailey, J. and Zha, H. and Song, L. and Xia, S.: Iterative learning with open-set noisy labels. In: CVPR. (2018)
38. Welinder, P. and Branson, S. and Perona, P. and Belongie, S.: The multidimensional wisdom of crowds. In: NeurIPS. (2010)
39. Yan, Y. and Rosales, R. and Fung, G. and Subramanian, R. and Dy, J.: Learning from multiple annotators with varying expertise. MLJ. (2014)
40. Yu, X. and Liu, T. and Gong, M. and Batmanghelich, K. and Tao, D.: An efficient and provable approach for mixture proportion estimation using linear independence assumption. In: CVPR. (2018)
41. Zhang, C.Y. and Bengio, S. and Hardt, M. and Recht, B. and Vinyals, O.: MentorNet: Understanding deep learning requires rethinking generalization. In: ICLR. (2017)
42. Zhang, Z. and Sabuncu, M.: Generalized cross entropy loss for training deep neural networks with noisy labels. In: NeurIPS. (2017)