IEG: ROBUST NEURAL NETWORK TRAINING TO TACKLE SEVERE LABEL NOISE

Anonymous authors
Paper under double-blind review

ABSTRACT

Collecting large-scale data with clean labels for supervised training of neural networks is practically challenging. Although noisy labels are usually cheap to acquire, existing methods suffer severely for training datasets with high noise ratios, making high-cost human labeling a necessity. Here we present a method to train neural networks in a way that is almost invulnerable to severe label noise by utilizing a tiny trusted set. Our method, named IEG, is based on three key insights: (i) Isolation of noisy labels, (ii) Escalation of useful supervision from mislabeled data, and (iii) Guidance from small trusted data. On CIFAR100 with a 40% uniform noise ratio and 10 trusted labeled data per class, our method achieves $80.2\pm0.3\%$ classification accuracy, only 1.4% higher error than a neural network trained without label noise. Moreover, increasing the noise ratio to 80%, our method still achieves a high accuracy of $75.5\pm0.2\%$, compared to the previous best 47.7%. Finally, our method sets new state of the art on various types of challenging label corruption types and levels and large-scale WebVision benchmarks.

1 INTRODUCTION

Training deep neural networks usually requires large-scale labeled data. However, the process of data labelling by humans is challenging and expensive in practice, especially in domains where expert annotators are needed such as medical imaging. A great number of methods have been proposed to train neural networks from datasets with noisy labels due to cheap acquisition (e.g. loosely-controlled procedures, crowd-sourcing, web search, text extraction, etc) (Zhang & Sabuncu, 2018). However, deep neural networks have high capacity for memorization. When noisy labels become prominent, neural networks inevitably overfit to noisy labeled data (Zhang et al., 2017a; Tanaka et al., 2018).

To overcome this problem, we argue that rethinking training dataset construction along with model training is necessary. Most methods consider a setting where the entire training dataset is acquired using the same labeling technique. However, when real-world constraints such as the labeling budget are considered, it is often practically feasible to also construct a tiny dataset that contains highly-trusted clean labels. If methods based on this setting can demonstrate high robustness even with extremely noisy labels, new horizons can be opened in training data labeling practices. There are a few recent methods that demonstrate strong performance by leveraging a small trusted dataset and training on a large noisy dataset, including learning weights of training data (Jiang et al., 2018; Ren et al., 2018), loss correction (Hendrycks et al., 2018), and knowledge graph (Li et al., 2017b). However, these methods still require a substantially large trusted set to reliably yield high performance. We show that it is possible to significantly reduce the necessary size of the trusted set while maintaining superior performance when suitable regularization is used (e.g. some methods use 10% of the total training data while our method only uses 0.2%).

In this paper, we consider three key factors and demonstrate a new method towards a noise robust neural net training strategy:

- **Isolation**: Reweigh training samples to isolate noisy labeled data and prevent mislabeled data from misleading neural network training.
- **Escalation**: Escalate supervision from mislabeled data via pseudo labels to make use of information in mislabeled data.

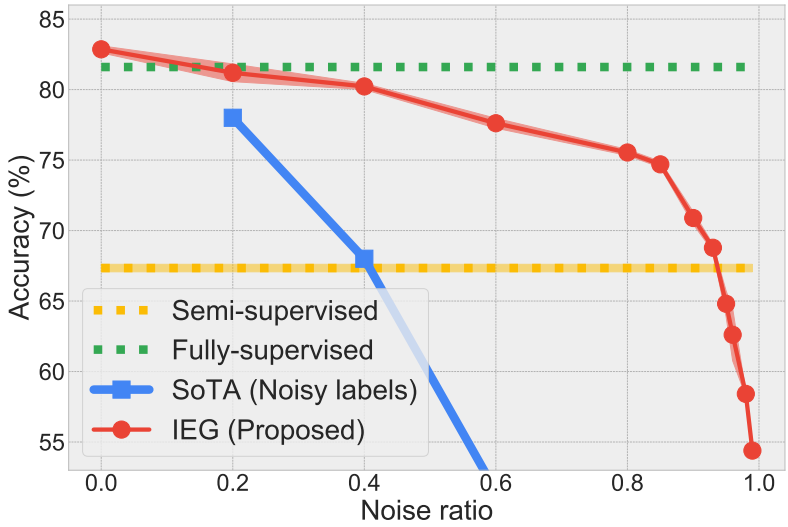

Figure 1: Image classification results on CIFAR100 showing the benefit of IEG. IEG denotes our method which outperforms semi-supervised learning methods at up to a 95% noise ratio. Fully-supervised is trained with all labeled clean data. Semi-supervised is our extension of IEG for semi-supervised setting, which has the best reported results. SoTA (noisy labels) denotes the previous best results for noise robustness (50 trusted data per class are used). The noise ratio of random label assignment is 0.99. 10 trusted labeled data per class are available for Semi-supervised and IEG. See Section 3.4 for more details.

- **Guidance**: Use a tiny trusted dataset for guided training with strong regularization to prevent overfitting.

Although previous work has attempted to deal with some of these factors, the performance gains have been moderate. Ideally, even with a small amount of correct labels in the noisy training data, a robust learning method should distill that information into model training and outperform the semi-supervised learning scenario where labels are completely ignored. However, from a view of comparison to state-of-the-art semi-supervised learning methods (Verma et al., 2019; Berthelot et al., 2019) in Figure 1 (explained in experiments), we observe that state of the art in noise-robust learning is inferior even with a 50% noise ratio (i.e. they cannot optimally distill the valuable supervised signal from almost half of the data), suggesting there is significant amount of room for improvement. This raises two important questions: 1. Should we discard noisy labels and opt in for semi-supervised training at high noise regimes? 2. How can we better distill the useful knowledge in noisy labels?

Contributions: First, we present a novel training method, named IEG, that tackles the above two questions effectively in an unified framework. IEG is designed to be model-agnostic and to generalize to any type of label corruption. Figure 1 demonstrates that even with extremely noisy labels (as high as 95%), our method is almost invulnerable to severe noise. We achieve the goal by addressing the three key factors effectively with the following complementary objectives:

1. A meta learning based re-weighting and re-labelling objective to simultaneously learn to weigh the per-datum importance and progressively escalate supervised losses of training data using pseudo labels as a replacement of original labels.

2. A label estimation objective to serve as the initialization of the meta re-labelling step and escalate supervision from mislabeled data.

3. An unsupervised regularization objective to enhance label estimation and improve overall representation learning.

Second, our method sets new state of the art on CIFAR10, CIFAR100 and on the large-scale WebVision, in many label corruption types by a large margin.

2 METHOD

2.1 META OPTIMIZATION-BASED REWEIGHING AND RELABELING

We leverage the meta optimization to automatically 1) estimate the weight of each data point when it is possibly mislabeled and 2) choose between pseudo labels and original labels when pseudo labels make useful contributions to the trusted set performance.

Given a dataset of N inputs with noisy labels $D_u = \{(x_i, y_i), 1 < i < N\}$ and also a small dataset (denoted as probe data) of M of samples with trusted labels $D_p = \{(x_i, y_i), 1 < i < M\}$, where $M \ll N$. The main idea of learning-to-reweight (L2R) (Ren et al., 2018) is training neural networks with a weighted cross-entropy loss for each training batch of size B:

$$\Theta^*(\omega) = \arg\min_{\Theta} \sum_{i=1}^{N} \omega_i L(y_i, \Phi(x_i; \Theta)), \tag{1}$$

where ω is a vector that its element ω_i gives the weight for the loss of a training pair. $\Phi(\cdot; \Theta)$ is the targeting neural network that outputs the class probability and $\sum_{i=1}^{N} L(y_i, \Phi(x_i; \Theta))$ is the standard softmax cross-entropy loss for each training data pair (x_i, y_i). Note that Θ is a function of ω, but we omit frequently for conciseness.

Treating ω as learnable parameters, the meta step behaves like a probe to seek for the optimal ω for each training data in D_u such that the trained model using equation 1 can obtain the best performance on the trusted data D_p. However, it is computationally-costly to find optimal ω^* since each update step requires training the model until converge to get Θ. In practice, we can use an online approximation (Ren et al., 2018; Finn et al., 2017) to perform a single meta gradient-descent step $\Theta_{t+1} = \Theta_t - \alpha \nabla_\Theta \sum_i^N \omega_i L\big(y_i, \Phi(x; \Theta_t)\big)$, where α is the step size,

$$\omega_t^* = \arg\min_{\omega, \omega \geq 0} \frac{1}{M} \sum_{i}^{M} L^p\big(y_i, \Phi(x_i; \Theta_{t+1}(\omega))\big), \ s.t. \sum_j \omega_{t,j} = 1. \tag{2}$$

We expect that the optimized ω^* should assign almost-zero value to mislabeled data to isolate mislabeled data from clean data. Optimization of ω is based on back-propagation with second-order derivatives.

When the noise ratio is high, a significant amount of data would be discarded. To address this information loss, we propose to generalize the meta update step to use information from the discarded data through a pseudo-labeling strategy. Given the function of pseudo label estimator $g(x, \Phi)$ (introduced in the next section), we generalize the meta optimization with the following objective to utilize both the data weighting and re-labelling:

$$\Theta^*(\omega, \lambda) = \arg\min_{\Theta} \sum_{i=1}^{N} \omega_i L\big(\lambda_i y_i + (1 - \lambda_i) g(x_i, \Phi), \ \Phi(x; \Theta_{t+1})\big). \tag{3}$$

The optimized re-labelling controller λ_t^* is updated based on the sign of its gradient,

$$\lambda_{t,i}^* = \text{sign}\Big(- \frac{\partial}{\partial \lambda_{t,i}} \mathbb{E}\big[L^p|_{\lambda=\lambda_0, \omega=\omega_0}\big]\Big), \tag{4}$$

where β is the step size. L^p is computed on a batch sampled from D_p. The reweighing controller $\omega^* = \omega - \nabla\omega$ can be obtained in the similar way. We use the sign of the gradient $|\nabla\lambda|$ instead of $\lambda_0 - \nabla\lambda$ because 1) $\nabla\lambda$ would get very small when pseudo labels are close to real labels (please see Appendix) and 2) simply averaging y_i and $g(x_i, \Phi)$ using scalar λ^* makes resulting pseudo label distribution less sharp.

After the meta step, we compute two cross-entropy losses given respective optimal values,

$$L_{\omega^*} = \sum_{i}^{N} \omega_i^* L\big(\lambda_0 y_i, \ \Phi(x_i; \Theta_t)\big), \ \ L_{\lambda^*} = \frac{1}{N} \sum_{i}^{N} L\big(\tilde{y}_i, \ \Phi(x_i; \Theta_t)\big), \tilde{y} = \begin{cases} y_i, & \text{if } \lambda_{t,i}^* > 0 \\ g(x_i, \Phi), & \text{otherwise} \end{cases} \tag{5}$$

where B is the batch size. Similar to L2R, we use momentum SGD for model training. We compute the meta step model parameters Θ_{t+1} by calculating the exact momentum update using momentum value of the SGD optimizer at each optimization step[1].

[1]We set initial the values as $\omega_0 = 1/B$ and $\lambda_0 = 0.9$ due to better performance, observed empirically.

2.2 ESCALATING SUPERVISION FROM MISLABELED DATA

Given learned data weights ω at a training step, IEG separates the data as either possibly-mislabeled or possibly-clean using the binary criterion $\mathbb{I}(\omega_i < T)$, where T is a scalar threshold. IEG utilizes mislabeled data with pseudo labels and probe data with trusted labels to provide training supervision with extra regularization, in order to achieve promised Escalation and Guidance.

2.2.1 PSEUDO LABELS

Utilizing the pseudo labels from unlabeled training data is widely studied for semi-supervised learning, by converting predictions to one-hot label (Lee, 2013) or their smoother versions (Tanaka et al., 2018; Lee, 2013)).

Neural network predictions can be unstable to input perturbations (Zheng et al., 2016; Azulay & Weiss, 2018). Enforcing consistency in neural network predictions has been shown to be important for model performance in semi-supervised learning (Xie et al., 2019). Therefore, if perturbations of an input obtain diverse model predictions, we should not trust the predictions to be pseudo labels. Recently-proposed state of the art semi-supervised learning method MixMatch (Berthelot et al., 2019) considers this principle in its design. We adopt this approach to compute $g(x_i, \Phi)$ in IEG. The resulting averaged predictions are given by: $g_i(x, \Phi) = Pr_i^{\frac{1}{\tau}} / \sum_i Pr_i^{\frac{1}{\tau}}$, where $Pr = \frac{1}{K}\big(\Phi(x) + \sum_k^{K-1} \Phi(\hat{x}_k)\big)$, where $\hat{x}_k$ is k-th randomly augmented version of input x. g leads to a soft pseudo label. g_i is i-th class of the pseudo label. τ is a softmax temperature scaling factor ($\tau = 0.5$ in this paper).

2.2.2 REGULARIZATION TO ENABLE GUIDANCE USING PROBE DATA

When noise ratio is high, even though the meta step effectively prevents misleading optimization (i.e. most elements in ω are zero), we potentially waste a lot of useful supervision to maintain high-performance learning. Therefore, we seek extra ways to improve supervision. Here, we show that even very small amount of trusted labeled data can improve model performance significantly. We aim to leverage the information from probe data, besides its original use for meta optimization. An appropriate regularization is critical for this purpose, otherwise the neural network would quickly overfit to the small probe data and yield ineffective meta gradients for learning ω and λ (e.g. equation 4).

To this end, we adopt the MixUp regularization and construct extra supervision losses using the data in the form of convex combinations using the data and their labels given a mixup factor β: $\text{Mix}_\beta(a, b) = \beta a + (1 - \beta)b, \ \beta \sim \text{Beta}(0.5, 0.5)$. It has been shown effective in recent semi-supervised learning methods (Hataya & Nakayama, 2019; Verma et al., 2019). In detail, for each data x_a in the concatenated data pool in $D_p \cup \hat{D}_u \cup D_u$, we apply pairwise MixUp between the input batch and its random permutation,

$$x_\beta = \text{Mix}_\beta(x_a, x_b),\ y_\beta = \text{Mix}_\beta(y_a, y_b), \ \text{ where } \{(x_a, y_a), (x_b, y_b) \in D_p \cup \hat{D}_u \cup D_u\}, \tag{6}$$

where $\hat{D}_u$ is the augmented copy of D_u. We introduce two softmax cross-entropy losses, L_β^p for resulting mixed data when $x_a \sim D_p$ is from probe data) and L_β^u when $x_a \sim \hat{D}_u \cup D_u$. We show that this strategy of IEG reduces the probe data size to be as small as one sample per class.

2.2.3 PSEUDO LABELS NEED CONSISTENT PREDICTIONS

Ideal pseudo labels should be close to real labels. The pseudo labels of IEG are generated by averaging the predictions over augmentations. However, if the predictions are controversial to each other, their contributions would cancel out and yield flattened average outputs. Consequently, the supervision using these pseudo labels would not encourage the model to be discriminative. In our insight, to generate more sharper pseudo outputs, reducing the controversy of augmentations is necessary. Therefore, we propose to encourage discriminability of the pseudo labels via enforcing consistency. The KL-divergence objective IEG used is defined as

$$\min_{\Theta} \ L_{KL} = \frac{1}{|D_u|} \sum_i^{|D_u|} \text{KL}\big(\Phi(x_i) \,\big|\big|\, \Phi(\hat{x}_i)\big). \tag{7}$$

Algorithm 1 summarizes a training step and presents all objectives along with their loss coefficients.

Algorithm 1: A training step of IEG at time step t

Input: Current model parameters Θ^t, A batch of training data X_u from D_u, a batch of probe data X_p from D_p, loss weight k and p, threshold T

Output: Updated model parameters Θ^{t+1}

1 Generate the augmentation $\hat{X}_u$ of X_u.

2 Estimate the persudo labels via $g(x_u, \Phi), x_u \sim X_u \cup \hat{X}_u$ (Section 2.2.1).

3 Compute optimal λ^* and ω^* via the meta step (Section 2.1).

4 Split the training batch X_u (also corresponding $\hat{X}_u$) to possible clean batch X_u^c and possible mislabeled batch X_u^u using the binary criterion $\mathbb{I}(\omega^* < T)$.

5 Compute the mixup of joint batch set (Section 2.2.2),

$$X_p \cup X_u^u \cup X_u^c \cup \hat{X}_u^u \cup \hat{X}_u^c$$

where $\hat{X}_u^u \cup X_u^u$ uses persudo labels estimated by $g(\cdot, \Phi)$.

6 Compute the total loss for model update

$$L_{\omega^*} + L_{\lambda^*} + L_\beta^p + p\, L_\beta^u + k\, L_{\text{KL}},$$

.

7 Conduct one step stochastic gradient descent to obtain Θ^{t+1}.

3 EXPERIMENTS

We validate the proposed IEG method on multiple datasets (CIFAR10, CIFAR100, and large-scale WebVision datasets) with various kinds of common label corruptions (including uniform and semantic types). We also conduct extensive ablation studies to demonstrate the key aspects of IEG.

3.1 EMPIRICAL TRAINING DETAILS

Here we discuss key training details and hyperparameters, that are shown to be beneficial for our experiments.

Learning rate decay: We adopt the Cosine learning rate decay with warm restarting[2]. We observe 3%-5% accuracy improvement on CIFAR datasets, especially at large noise ratios. Figure A1 of Appendix plots the curves. Although it works particularly well in IEG, we do not observe strong benefit either training standard neural networks or training L2R, also not in recent literature (Gotmare et al., 2019) and (Song et al., 2018) which uses cosine learning rate.

Model selection: Although the size of probe data is small, we find our method less likely memorizes the probe data. So it can be potentially be monitored as validation set for model selection. Loshchilov & Hutter (2017) also indicates the needless of the validation set for model selection with cosine learning rate decay. Therefore, we directly select models at the lowest learning rate before 200 epochs.

Augmentation: The purpose of augmentation is to generate pixel perturbation around the original training data. We adopt the AutoAugment (AA) technique for image data (Cubuk et al., 2018) to achieve this, which including operations (learned policy augmentation →flip→random crop→cutout (DeVries & Taylor, 2017)). In detail, for each input image, we first generate one standard augmentation (random crop and horizontal flip) and then apply AA to generate K random augmentations on top of the standard one. We use $K = 2$ augmentations in our experiments.

3.2 CIFAR NOISY LABEL EXPERIMENTS

For all CIFAR experiments, we set $T = 1, p = 5, k = 20$. The models are trained on a single NVIDIA v100. std of reported results are obtained by 3 runs with random seeds. We compare the proposed IEG method against several recent methods, which achieve leading performance in public

[2]We set the initial cycle length to be one epoch, and after then cycle length increases by a factor of 1.5 and meanwhile the restart learning rate decreases by a factor of 0.9 as described in (Loshchilov & Hutter, 2017)

Method	$\|D_p\|$	Noise ratio			
		0	0.2	0.4	0.8
GCE (Zhang & Sabuncu, 2018)	-	93.5	89.9±0.2	87.1±0.2	67.9±0.6
MentorNet DD (Jiang et al., 2018)	5k	96.0	92.0	89.0	49.0
RoG (Lee et al., 2019)	-	94.2	87.4	81.8	-
L2R (Ren et al., 2018)	1k	96.1	90.0±0.4*	86.9±0.2	73.0±0.8*
(Arazo et al., 2019)	-	-	93.8	92.3	74.1
IEG	0.1k	96.8	**96.2±0.2**	**95.9±0.2**	**93.7±0.5**
IEG-RN29	0.1k	94.4	92.9±0.2	92.5±0.5	85.6+1.1

Table 1: Validation accuracy on CIFAR10 with uniform noise. $|D_p|$ denotes the number of trusted (probe) data used. 0.1k indicates 10 images per class. For reference, standard training of WRN-28-10/ResNet29 (RN29) leads to 96.1%/92.7% accuracy. * indicates results trained by us.

Method	$\|D_p\|$	Noise ratio			
		0	0.2	0.4	0.8
GCE (Zhang & Sabuncu, 2018)	-	81.4	66.8±0.4	61.8±0.2	47.7±0.7
MentorNet DD (Jiang et al., 2018)	5k	79.0	73.0	68.0	35.0
L2R (Ren et al., 2018)	1k	81.2	67.1±0.1*	61.3+2.0	35.1±1.2*
(Arazo et al., 2019)	-	-	70.0	64.4	45.5
IEG-RN29	1k	70.3	69.3±0.5	67.0±0.8	60.7±1.0
IEG	0.1k	83.0	77.4±0.4	75.1±1.1	62.1±1.2
IEG	0.5k	83.0	80.4±0.5	79.6±0.3	73.6±1.5
IEG	1k	83.0	**81.2±0.7**	**80.2±0.3**	**75.5±0.2**

Table 2: Validation accuracy on CIFAR100 with uniform noise. Standard training of WRN-28-10/RN29 leads to 81.6%/71.3% accuracy. 0.1k indicates 1 images per class.

Method	Noise ratio		
	0.2	0.4	0.8
GCE	89.5±0.3	82.3±0.7	-
LC	89.1±0.5	83.6±0.3	-
IEG-RN29	92.7±0.2	90.2±0.5	78.9±3.5
IEG	**96.5±0.2**	**94.9±0.1**	**79.3±2.4**

Table 3: Asymmetric noise on CIFAR10. LC is a loss correction approach (Patrini et al., 2017). 10 trusted data per class are used as probe data.

Method	C10 (34%)	C100 (37%)
RoG	70.0	53.6
L2R	71.0*	56.9*
IEG-RN29	81.8	65.1
IEG	**88.3**	**73.7**

Table 4: Semantic noisy experiments where labels are generated by a neural network on a few data. Noise ratio is shown in parentheses. RoG uses DenseNet-100.

benchmarks. Similar to L2R, we use the Wide ResNet (WRN28-10) (Zagoruyko & Komodakis, 2016) as default, unless specified otherwise, for fair comparison.

Common random label noise: Table 1 compares the results for CIFAR10 with uniform noise ratios of 0.2, 0.4, and 0.8. 10 probe images per class are used. We also test IEG using ResNet29[3], which is much smaller than ones used by compared methods. Using WRN28-10, IEG leads to 96.5% accuracy with 20% noise ratio and 94.7% accuracy with 80% noise ratio, demonstrating nearly noise-free performance. IEG still achieves the best performance with ResNet29. We also train IEG with 0% noise as reference. We observe most results even outperforms the results with standard training of WRN28-10/ResNet29 (see captions of Table 1). This shows that our proposed method provides additional form of regularization to improve generalization. Table 2 compares the results in CIFAR100 with uniform noise ratios of 0.2, 0.4, and 0.8. We also report results given 10 images, 5 images and the extreme case of 1 image per class for probe data, much lower than the other methods use. IEG significantly outperforms existing methods. The improvement is remarkable at higher noise ratios.

Three types of semantic label noise: Next, we test IEG on more realistic noisy settings on CIFAR. 10 images per class are used as probe data. Table 3 compares the results on CIFAR10 with asymmet-

[3]For ResNet29 we use in this paper, we follow this pre-activation (v2) implementation `https://github.com/keras-team/keras/blob/master/examples/cifar10_resnet.py`, which contains 0.84M parameters.

Dataset	ResNet-50	(Chen et al., 2019)	MentorNet	IEG-RN50
mini	61.0/84.3	61.6/85.0	63.8/85.8	**72.6/91.5**
full	57.2/79.3	-	64.2/84.8	**65.8/85.8**

Table 5: Large-scale WebVision experiments. The top-1/top-5 accuracy on the ImageNet validation set are compared. IEG uses ResNet-50. The full version does not use AA.

Dataset	MixMatch	EG	EG*	IEG
CIFAR10	51.2	92.4±0.7	94.5±0.3	93.7±0.5
CIFAR100	34.5	57.6±0.4	67.3±0.3	75.2±0.2

Table 6: Comparison with semi-supervised methods. MixMatch and EG use WRN-28-2. EG* and IEG use WRN-28-10. 10 labeled data per class are used. The same size is used for probe data in IEG. The results of IEG are reported under the 80% uniform noise ratio.

ric noise ratios of 0.2, 0.4, and 0.8. Asymmetric noise is known as a more realistic setting because it corrupts semantically-similar classes (e.g. truck and automobile, bird and airplane) (Patrini et al., 2017). Moreover, we follow RoG (Lee et al., 2019) to generate semantic noisy labels by using a trained VGG-13 (Simonyan & Zisserman, 2015) (the hardest setting) on 5% of CIFAR10 and 20% of CIFAR100 (we directly use the data provided by the author). Table 4 reports the compared results. Lastly, we test IEG on three kinds of open-set noisy labels (this setting replaces images to out-of-distribution images of the same labels (Wang et al., 2018)) in Table A1 of Appendix. In all semantic noise settings, IEG consistently outperforms the compared methods by a significant margin.

3.3 WEBVISION REAL-WORLD NOISY LABEL EXPERIMENTS

WebVision (Li et al., 2017a) is large-scale dataset which reflects real-world noisy labels as their images are obtained by crawling from the Flickr website and Google Images Search using the labels. It contains 2.4 million images and shares the 1000 classes with ImageNet (Deng et al.). We also follow (Jiang et al., 2018) to create a mini version of WebVision, which includes the Google subset images of the top 50 classes. To create the probe data, we set aside 10 images per class from the ImageNet training data. We train models using the WebVision training set on a Google Cloud TPU and evaluate on the ImageNet validation set. We set $T = 1, p = 4, k = 8$. Table A2 compares the results. While the compared methods use a larger InceptionResNetv2 (IRv2), we use ResNet-50 due to its memory efficiency, albeit the lower expected performance due to its lower capacity. We verify the performance of backbone neural networks. We observe slight (<0.5%) gain when we test baseline ResNet-50 by adding the probe data in training (reported in the table). AA is effective but time consuming. We use standard image augmentation instead of AA for the full Webvision. On mini, standard training ResNet-50 leads to 52.6/84.3 without AA. Standard training IRv2 leads to 57.2/79.2 without AA and 64.0/84.2 with AA.

3.4 COMPARISON TO SEMI-SUPERVISED LEARNING

We compare IEG to the state-of-the-art semi-supervised learning method MixMatch Berthelot et al. (2019) to verify how much useful information IEG can distill from mislabeled data. The unsupervised components which IEG incorporated can be also applied onto semi-supervised methods. We simply remove the Isolation (meta re-weighting and re-labelling) of IEG and treat all training data with noisy labels as unlabeled to enable semi-supervised training (denoted the resulting method as EG). Figure 1 shows the comparisons and Table 6 reports the detailed results. IEG improves the performance largely given the 80% label noise ratio. In addition, EG demonstrates remarkable benefits, for example, from 34.5% to 57.6% on CIFAR100. It demonstrates that it is necessary to enforce consistency for pseudo labels for better discriminability.

3.5 ABLATION STUDIES AND DISCUSSIONS

Here we study the individual objective components of IEG and their effectiveness. Table 7 summarizes the ablation study results (referred as IEG-#) and we discuss them further below.

Unsupervised consistency (UC): Based on our empirical observations, UC plays an important role in preventing neural networks from overfitting to samples with wrong labels, especially at extreme noise ratios. IEG-4 shows results without UC. Figure A2 in Appendix shows the training curves with different coefficient k for L_{KL}. At around 80k iteration, the curve of $\beta = 1$ starts to overfit to

#	Component (abbr.)				Noise ratio	
	UC	MC	AA	λ	0.4	0.8
1					64.43	33.52
2				✓	66.14	36.04
3			✓	✓	67.82	37.01
4		✓	✓	✓	78.06	61.81
5	✓	✓	✓	✓	79.96	75.42
6		✗			73.63	54.76
7			✗		79.16	72.69
8				✗	81.05	74.04

Table 7: Ablation study on CIFAR100. ✓/✗ indicates the corresponding component is enabled/disabled. So IEG-1 is equal to L2R; IEG-5 is the full IEG. Abbreviations are defined in text.

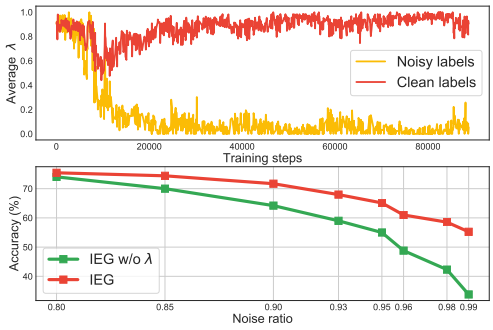

Figure 2: Analysis of λ. Top: The average λ of noisy and clean labels on CIFAR10 with 40% noise. Bottom: Accuracy (w/o λ) at extreme noise ratios on CIFAR100.

noisy labels and simultaneously the validation accuracy starts to decrease. $\beta = 20$ is much more efficient in overcoming this.

The effects of input perturbation: IEG-7 shows the results after removing AA-learned policy augmentation (we only use flip $\rightarrow$ random crop $\rightarrow$ cutout). Cutout is effective, as also observed in (Xie et al., 2019). Removing it leads to 72.71%/62.41% for 40%/80% noise.

Mixup cross-entropy (MC) regularization: Directly minimizing the cross-entropy loss of the tiny probe data would make the model quickly memorize them. IEG-6 shows the result without MC regularization. The performance loss is significant at 80% noise ratio. Therefore, MC effectively brings useful supervision in probe data to guide training.

The effects of λ: Our proposed meta re-labeling (equation 3) is very effective for extremely-high noise ratios. It learns to assign lower λ for mislabeled data in order to promote the use of pseudo labels, and vice visa for clean data. Figure 2 (top) shows the average λ during the training process (the value of noise labels are obtained by peeping ground truth). Figure 2 (bottom) demonstrates the significant advantage of λ under extreme noise ratios.

4 Related Work

Reweighing training data has been shown to be effective (Liu & Tao, 2015). However, estimating effective weights is challenging. Ren et al. (2018) proposes a meta learning approach to directly optimize the weights in pursuit of best validation performance. Jiang et al. (2018) alternatively uses teach-student curriculum learning to weigh data. (Han et al., 2018) uses two neural networks to co-train and feed data to each other selectively. Another direction is modeling confusion matrix for loss correction, which has been widely studied (Sukhbaatar et al., 2014; Natarajan et al., 2013; Tanno et al., 2019; Patrini et al., 2017; Arazo et al., 2019). For example, (Hendrycks et al., 2018) shows that using a set of trusted data to estimate the confusion matrix has significant gains.

The approach on relabeling corrupted samples is another direction (Li et al., 2017b; Tanaka et al., 2018; Veit et al., 2017; Han et al., 2019). Along this, Reed et al. (2014) uses bootstrapping to generate new labels. (Li et al., 2019) leverage the meta learning framework to verify multiple label candidates before doing actual training. Relabeling is similar to the pseudo label approach in semi-supervised learning (Lee, 2013). Besides pseudo labels, building connections to semi-supervised learning has been recently expanded (Kim et al., 2019), which applies semi-supervised losses to improve representation learning from mislabeled data. For example, Hataya & Nakayama (2019); Arazo et al. (2019) uses Mixup (Zhang et al., 2017b) to augment data and demonstrates clear benefits. Ding et al. (2018); Kim et al. (2019) identifies mislabeled data first and then leverages semi-supervised techniques.

5 Conclusion

In this paper, we present a robust and generic neural network training method to overcome severe label noise. Our method, named IEG, is based on unification of the mechanisms to isolate the noise labels via meta optimization, escalate the supervision mislabeled data via pseudo meta re-labeling, and effectively use small trusted data to guide training. IEG demonstrates significant and consistent improvements over previous state of the art methods on common benchmarks.

ACKNOWLEDGMENTS

We would like to thank Liangliang Cao, Kihyuk Sohn, David Berthelot, Qizhe Xie, and Chen Xing for their valuable discussions.

REFERENCES

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

Aharon Azulay and Yair Weiss. Why do deep convolutional networks generalize so poorly to small image transformations? *arXiv preprint arXiv:1805.12177*, 2018.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. *arXiv preprint arXiv:1905.02249*, 2019.

Pengfei Chen, Benben Liao, Guangyong Chen, and Shengyu Zhang. Understanding and utilizing deep neural networks trained with noisy labels. *arXiv preprint arXiv:1905.05040*, 2019.

Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*, 2018.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255.

Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

Yifan Ding, Liqiang Wang, Deliang Fan, and Boqing Gong. A semi-supervised two-stage approach to learning from noisy labels. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1215–1224, 2018.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning (ICML)*, pp. 1126–1135, 2017.

Akhilesh Gotmare, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. A closer look at deep learning heuristics: Learning rate restarts, warmup and distillation. *International Conference on Learning Representations (ICLR)*, 2019.

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. *Advances in neural information processing systems (NeurIPS)*, 2018.

Jiangfan Han, Ping Luo, and Xiaogang Wang. Deep self-learning from noisy labels. *IEEE International Conference on Computer Vision (ICCV)*, 2019.

Ryuichiro Hataya and Hideki Nakayama. Unifying semi-supervised and robust learning by mixup. 2019.

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

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. *International Conference on Machine Learning (ICML)*, 2018.

Youngdong Kim, Junho Yim, Juseung Yun, and Junmo Kim. Nlnl: Negative learning for noisy labels. *International Conference on Computer Vision*, 2019.

Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, volume 3, pp. 2, 2013.

Kimin Lee, Sukmin Yun, Kibok Lee, Honglak Lee, Bo Li, and Jinwoo Shin. Robust inference via generative classifiers for handling noisy labels. *International Conference on Machine Learning (ICML)*, 2019.

Junnan Li, Yongkang Wong, Qi Zhao, and Mohan S Kankanhalli. Learning to learn from noisy labeled data. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5051–5059, 2019.

Wen Li, Limin Wang, Wei Li, Eirikur Agustsson, and Luc Van Gool. Webvision database: Visual learning and understanding from web data. *arXiv preprint arXiv:1708.02862*, 2017a.

Yuncheng Li, Jianchao Yang, Yale Song, Liangliang Cao, Jiebo Luo, and Li-Jia Li. Learning from noisy labels with distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1910–1918, 2017b.

Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *Transactions on pattern analysis and machine intelligence*, 38(3):447–461, 2015.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *International Conference on Learning Representations (ICLR)*, 2017.

Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *Advances in neural information processing systems (NeurIPS)*, pp. 1196–1204, 2013.

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

Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014.

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

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations (ICLR)*, 2015.

Jiaming Song, Tengyu Ma, Michael Auli, and Yann Dauphin. Better generalization with on-the-fly dataset denoising. 2018.

Sainbayar Sukhbaatar, Joan Bruna, Manohar Paluri, Lubomir Bourdev, and Rob Fergus. Training convolutional networks with noisy labels. *arXiv preprint arXiv:1406.2080*, 2014.

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

Ryutaro Tanno, Ardavan Saeedi, Swami Sankaranarayanan, Daniel C Alexander, and Nathan Silberman. Learning from noisy labels by regularized estimation of annotator confusion. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Andreas Veit, Neil Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge Belongie. Learning from noisy large-scale datasets with minimal supervision. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 839–847, 2017.

Vikas Verma, Alex Lamb, Juho Kannala, Yoshua Bengio, and David Lopez-Paz. Interpolation consistency training for semi-supervised learning. *arXiv preprint arXiv:1903.03825*, 2019.

Yisen Wang, Weiyang Liu, Xingjun Ma, James Bailey, Hongyuan Zha, Le Song, and Shu-Tao Xia. Iterative learning with open-set noisy labels. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8688–8696, 2018.

Qizhe Xie, Zihang Dai, Eduard H. Hovy, Minh-Thang Luong, and Quoc V. Le. Unsupervised data augmentation. *arXiv*, 2019.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *British Machine Vision Conference (BMVC)*, 2016.

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

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *International Conference on Learning Representations (ICLR)*, 2017b.

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

Stephan Zheng, Yang Song, Thomas Leung, and Ian Goodfellow. Improving the robustness of deep neural networks via stability training. In *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4480–4488, 2016.

APPENDIX

A MORE RESULTS

Here we show more analytically results and comparison results as being referred in the main text.

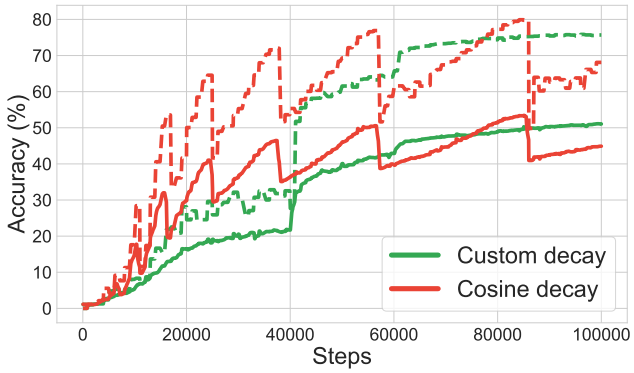

Figure A1: Compared with custom learning rate decay strategy. We use the commonly accepted setting (also used by L2R): the initial learning rate is 0.1, the learning rate decays to previous 0.1x at 40K and 50K steps. We show the training curves on CIFAR10 with 40% uniform label noise. Dotted and solid lines are training and evaluation accuracy curves, respectively.

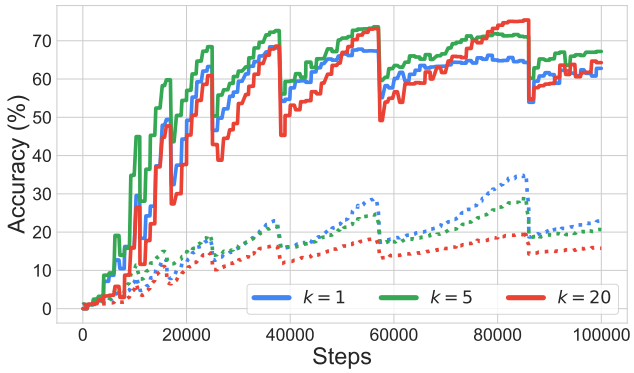

Figure A2: Training curves on CIFAR100 with uniform 80% label noise under different L_{KL} loss weight k (defined in Algorithm 1). Dotted are solid lines are train and evaluation accuracy curves, respectively. Since the noise ratio is 80%, the average training accuracy is expected to be lower than 20%, otherwise the model starts to overfit. When we use a small k, the model becomes to overfit at around 80000 iterations.

B PROOF OF SMALL $\nabla\lambda$

Here we show that the derivative of λ_t, $-\frac{\partial}{\partial\lambda_{t,i}}\mathbb{E}\left[L_p|_{\lambda=\lambda_0,\omega=\omega_0}\right]$ inside the sign function of equation 4 will be very small when pseudo labels are close to real labels.

Open-set noise type	CIFAR100	CIFAR100+ImageNet	ImageNet
RN29	77.8	80.34	84.43
DenseNet-100	79.0	86.7	81.6
WRN-28-10	82.8	84.7	88.7
L2R	81.8	81.3	85.0
RoG	83.4	87.1	84.4
IEG-RN29	86.4	87.4	90.0
IEG	**92.3**	**93.0**	**94.0**

Table A1: Open-set noise on CIFAR10. We follow the setting and created noisy datasets of RoG to conduct experiments. Each column indicates where the noisy out-of-distribution images are from. Three types of noisy types are compared. RoG uses DenseNet-100 and L2R use WRN-28-10. We run the baseline for better comparison (the first block of the table). From results of WRN-28-10, we can see model capacity is beneficial for performance. It is interesting to L2R does not outperforms the its backbone baseline WRN-28-10, which implies that only data reweighting is not effective to deal with open-set noise.

Ratio	0	0.2	0.4	0.6	0.8	0.85	0.9	0.93	0.95	0.96	0.98	0.99
mean	82.9	81.2	80.2	77.6	75.5	74.7	70.9	68.8	64.8	62.6	58.4	54.4
std	0.25	0.63	0.22	0.35	0.21	0.21	0.45	0.26	0.91	1.85	0.16	0.29

Table A2: Accuracy (mean and std) of IEG on CIFAR100 with different uniform noise ratios.

$$\frac{1}{M}\sum_{i=1}^{M}\frac{\partial}{\partial\lambda_{i,t}}L_p(y_i,\Phi(x_i;\Theta))|_{\omega_{i,t}=\omega_0,\lambda_{i,t}=\lambda_0} \tag{8}$$

$$=\frac{1}{M}\sum_{i=1}^{M}\frac{\partial L_p(y_i,\Phi(x_i;\Theta))}{\partial\Theta}|^T_{\Theta=\Theta_t}\frac{\partial\Theta_{t+1(\lambda_{i,t})}}{\partial\lambda_{i,t}}|_{\omega_{i,t}=\omega_0,\lambda_{i,t}=\lambda_0} \tag{9}$$

$$\propto\sum_{i=1}^{M}\frac{\partial L_p(y_i,\Phi(x;\Theta))}{\partial\Theta}|^T_{\Theta=\Theta_t}\frac{\partial(\omega_{i,t}\cdot L(y_i,\Phi(x_i;\Theta))-\omega_{i,t}\cdot L(g(\Phi(x_i;\Theta),\Phi(x_i;\Theta))}{\partial\Theta}|_{\Theta=\Theta_t,\omega_{i,t}=\omega_0} \tag{10}$$

$$\propto\sum_{i=1}^{M}\frac{\partial L_p(y_i,\Phi(x_i;\Theta))}{\partial\Theta}|^T_{\Theta=\Theta_t}\frac{\partial(L(y_i,\Phi(x_i;\Theta))-L(g(\Phi(x_i;\Theta),\Phi(x_i;\Theta))}{\partial\Theta}|_{\Theta=\Theta_t} \tag{11}$$

If y_i and $\Phi(x_i;\Theta)$ are close to each other around Θ_t, the derivative $\frac{\partial}{\partial\lambda_{i,t}}\mathbb{E}\big[L_p\big|_{\lambda=\lambda_0,\omega=\omega_0}\big]$ would be close to 0. Thus, for a converged model with low training error, the amount of update on λ would be close to zero.