A Survey on Deep Learning with Noisy Labels: How to train your model when you cannot trust on the annotations?

Filipe R. Cordeiro* and Gustavo Carneiro†
*Department of Computing, Federal Rural University of Pernambuco, Brazil
†School of Computer Science, Australian Institute for Machine Learning, University of Adelaide, Australia
Email: filipe.rolim@ufrpe.br, gustavo.carneiro@adelaide.edu.au

***Abstract*—Noisy Labels are commonly present in data sets automatically collected from the internet, mislabeled by non-specialist annotators, or even specialists in a challenging task, such as in the medical field. Although deep learning models have shown significant improvements in different domains, an open issue is their ability to memorize noisy labels during training, reducing their generalization potential. As deep learning models depend on correctly labeled data sets and label correctness is difficult to guarantee, it is crucial to consider the presence of noisy labels for deep learning training. Several approaches have been proposed in the literature to improve the training of deep learning models in the presence of noisy labels. This paper presents a survey on the main techniques in literature, in which we classify the algorithm in the following groups: robust losses, sample weighting, sample selection, meta-learning, and combined approaches. We also present the commonly used experimental setup, data sets, and results of the state-of-the-art models.**

I. Introduction

Deep Neural Networks (DNNs) have shown great performance to deal with different computer vision tasks, such as image classification [1], segmentation [2] and object detection [3], to different areas of applications [4]–[6]. One of the factors that improve the performance of deep learning models is the use of large-scale datasets, such as ImageNet [7]. Unfortunately, the labeling process of large-scale datasets is expensive and time-consuming, and researchers sometimes resort to cheaper alternatives, such as online queries [8] and crowdsourcing [9], which can produce datasets with incorrect or noisy labels. Incorrect labels may also be present in small datasets, where the labeling task is difficult or have divergent opinions between annotators, such as medical images [10], [11].

As stated in [12], noisy labels may occur naturally when human annotators are involved. Frenay et al. [13] summarize the main sources of noisy labels into four types: 1) insufficient information to provide reliable labeling, such as poor quality images; 2) mistake made by experts; 3) variability in the labeling by several experts; and 4) data encoding or communication problems (e.g., accidental click). Figure 1 illustrates the labeling strategies and sources of noisy labels.

Most of the solutions using DNN assume that either the labels were annotated by experts or were curated and therefore, would have been perfectly annotated. However, that is not a realistic assumption, mainly when dealing with data collected in an unsupervised way (eg., web queries). As a consequence, a DNN trained with noisy labels might decrease the accuracy and require larger training sets [14].

Noisy labels are also related to semi-supervised learning. As observed by Wang et al. [15], when missing labels are incorrectly labeled in a semi-supervised approach, the challenge of semi-supervised learning approximates to noisy labels. The same can be said about pseudo-labeling techniques, where samples can be mislabeled. However, the noisy label problem is even more challenging than previous problems because we do not have information about which samples have clean labels.

Zhang et al. [14] have shown that Convolutional Neural Networks (CNNs) can easily fit any ratio of noisy labels, leading to poor generalization performance. However, it was shown that easy patterns, corresponding to easy samples with clean labels, are learned first, while difficult patterns, which are closer to noisy labels, are learned later. Based on this observation, many proposed methods are based on the *small-loss trick*, which consists of selecting the samples with small loss to be used as clean samples [16]–[18].

Most of the works proposed to deal with noisy labels try to answer the following questions: *How to identify the noisy samples?*, or more generally: *How to effectively train on noisy labeled datasets?* [19]. To address this problem, different strategies have been proposed: robust losses [20], [21], label cleansing [22], [23], weighting [24], meta-learning [25], ensemble learning [26], and others [9], [27], [28]. This survey describes the main approaches proposed in the literature related to training deep neural networks in the presence of noisy labels.

II. Label noise: Definition and Taxonomy

A. Definition

In this document, we refer to noisy samples as the ones whose labels are different from their true class. For these samples, we denote their labels as noisy labels, which means their labels are wrong. Therefore, when referring to noisy samples in the scope of this paper, it means that the noise is present only in the labels and not in the input data.

2377-5416/20/$31.00 ©2020 IEEE
DOI 10.1109/SIBGRAPI51738.2020.00010

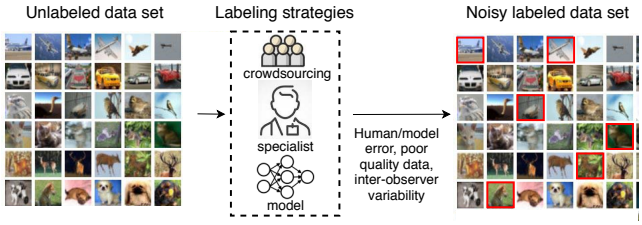


Fig. 1. Labeling process and noise sources.

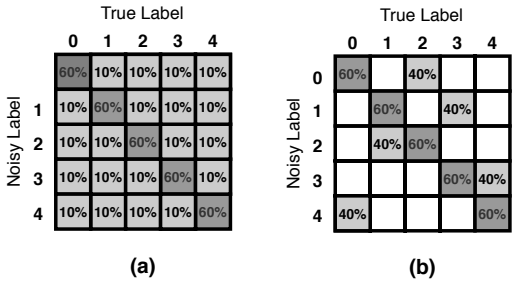


Fig. 2. Transition matrix of different noisy types: (a) symmetric, and (b) asymmetric, for $\eta = 0.4$ and 5 classes.

B. Problem statement

Lets consider a classification problem with a training set $D = \{(x_1, y_1), ..., (x_n, y_n)\}$, where $x_i \in \mathcal{X}$ is the i^{th} image and $y_i \in Y$ is a one-hot vector representing the label over c classes. In a noisy label scenario, the labels might be wrong and we denote $y \in Y$ as the observed labels, which may contain noise (i.e., incorrect labels). We denote the true label of x_i as y_i^*. We denote the distribution of different labels for sample x by $p(c|x)$, and $\sum_{c=1}^{C} p(c|x) = 1$.

A supervised classification learns a function $f : \mathcal{X} \rightarrow \mathcal{Y}$ that maps the input space to the label space. Training the classifier has as objective to find the optimal parameters θ that minimize an empirical risk defined by a loss function. Given a loss function, L, and a classifier, $f(.)$, the empirical risk is defined as follows:

$$R_L(f) = \mathbb{E}_{(x,y)\in D}[L(f(x), y)] = \mathbb{E}_{x,y_x}[L(f(x), y_x)], \quad (1)$$

where $\mathbb{E}$ denotes the Monte-Carlo expectation using the training set D.

C. Types of noise

We denote the overall noise rate by $\eta \in [0, 1]$ and η_{jc} represents the probability of a class j be flipped to class c, as $p(y_i = c|y_i^* = j)$. The main types of noise studied in literature are described as follow:

1) Symmetric Noise: Symmetric noise is also called random noise or *uniform* noise, and it represents a noise process when a label has equal probability to flip to another class. Among the symmetric noise definition, there are two variations: *symm-inc* and *symm-exc* [27]. In *symm-inc*, the true label is included into the label flipping options, which means that in $\eta_{jc} = \frac{\eta}{C-1}, \forall j \in Y$, while in the *symm-exc* the true label is not included, which means $\eta_{jc} = \frac{\eta}{C-1}, j \neq c$. The symmetric noise, or random noise, is unlikely to represent a realistic scenario for noisy labels, but it is the main baseline for noisy labels experiments. Figure 2 (a) shows the transition matrix for symmetric noise, with $\eta = 0.4$.

2) Asymmetric Noise: : The asymmetric noise, as described in [29], is closer to a real-world label noise based on flipping labels between similar classes. For example, using CIFAR10 dataset [30], the asymmetric noise maps TRUCK→ AUTOMOBILE, BIRD → PLANE, DEER → HORSE, as mapped by [31]. For MNIST, [29] maps $2 \rightarrow 7$, $3 \rightarrow 8$, $7 \rightarrow 1$ and $5 \rightarrow 6$. For asymmetric noise, η_{jc} is class conditional. Figure 2 (b) shows the transition matrix for asymmetric noise, with $\eta = 0.4$.

3) Open-set Noise: : The noisy label problem can fall into two categories: closed-set and open-set. A closed-set noisy problem is when all the true labels belong to the known classes. For example, for MNIST, if a subset of samples has the labels flipped to wrong labels, their original images, and corresponding true classes belong to MNIST.

The open-set noise is when a sample has a true class that is not contained in the known classes of the training data. For example, if an image from CIFAR is contained in MNIST training with an incorrect label of 7, it will be trained as a class 7, but it would be an open-set sample. This type of noise is commonly found when obtaining images from automatic web search engines (e.g., Google Images).

III. Literature Review

Learning in the presence of noisy labels is a problem studied during the last decades [32], and several strategies have been proposed to make models more robust to noise [13]. In this work, we are grouping the main approaches in the area in the following groups: noise transition matrix, robust losses, sample weighting, sample selection, meta-learning, and combined approaches. Each category is described below:

A. Noise Transition Matrix

Most of the first approaches proposed to deal with noisy labels were based on estimating a noise transition matrix to learn how labels switch between classes, as illustrated in Figure 2. The cross-entropy loss with transition matrix is defined as follows:

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{n=1}^{N} -log\ p(y = y_n|x_n, \theta), \quad (2)$$

where

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{n=1}^{N} -log(\sum_{i}^{c} p(y = y_n|y^* = i)p(y^* = i|x_n, \theta)). \quad (3)$$

Several methods have been proposed to estimate the transition matrix. Patrini et al. [29] estimate this matrix using a pre-trained model. Hendricks et al. [33] use a clean validation

set to calculate the transition matrix, while Sukhbaatar et al. [34] propose the use of the difference between the transition matrices calculated from clean and noisy data.

Reed et al. [35] uses a transition matrix combined with a regularized loss that uses a combination of the noisy labels and labels predicted by the model. Goldberger et al. [36] use the expectation-maximization (EM) algorithm to find the optimal parameters of both network and the noise. The use of transition matrices has been further explored in different ways [37]–[39].

B. Robust Losses

Loss correction approaches usually add a regularization or modify the network probabilities to penalize less the low confident predictions, which may be related to noisy samples. One of the advantages of these approaches is that they can be used with any model. Most of the methods treat the noisy and clean samples the same way, but penalise less the low confident prediction samples, compared to the standard cross-entropy [3].

Manwani et al. [40] show that 0-1 losses are more noise-tolerant than commonly used convex losses. [41] compare categorical cross-entropy (CCE) with mean absolute value of error (MAE) losses, and show that MAE is more noise tolerant because MAE treats all data points equally. However, training with MAE usually leads to underfit, and it may not be beneficial using it depending on the noise type. Zhang and Sabuncu [31] propose a generalized cross-entropy loss by combining the benefits of mean absolute error and cross-entropy losses. Wang et al. [20] propose an improved version of MAE (IMAE), which uses hyperparameters to adjust the weighting variance of MAE. Wang et al. [21] propose the symmetric cross-entropy, based on the fact that CCE is not symmetric. By adding symmetry to the loss, they show that it helps to deal with noisy labels. Ziyin et al. [42] propose the use of a loss function that encourages the model to abstain from learning samples with noisy labels. This idea is similar to re-weighting or sample selection, where the noisy samples can be observed as having zero weight or removed. Similarly, [43] also proposes a loss function that permits abstention during training. Although some approaches benefit from removing noisy samples, using them through relabelling or giving a lower weight has shown to improver results.

Ma et al. [44] propose a robust loss function called Active Passive Loss (APL) that combines two robust loss functions that mutually boost each other. Ma et al. identify that existing robust loss functions can deal with noisy labels, but suffer from the problem of underfitting. Their proposal addresses this problem by combining losses that cause overfit, such as CCE, with one that causes underfit, as MAE.

Liu et al. [45] propose a peer loss function inspired in a peer prediction mechanism. They show that peer loss functions on the noisy data lead to the optimal or a near-optimal classifier as if performing training over the clean training data.

C. Sample Weighting

Wang et al. [46] propose a weighting scheme to reduce the contribution of the noisy samples. Similarly, [47] uses a method based on Curriculum Learning, which defines a weight to each sample based on an unsupervised estimation of data complexity.

Xue et al. [48] use a probabilistic Local Outlier Factor algorithm (pLOF), which is used as an outlier detector, to estimate a probability value of a sample be an outlier (i.e., noisy sample). [46] also uses the pLOF, but combined with a Siamese Network training. Using siamese networks encourages the model to learn similar features between clean samples of the same class and different ones among clean and noisy samples.

Harutyunyan et al. [49] show that the memorization of label noise can be reduced by reducing the mutual information between weights. In their proposal, they update the weights based on the gradients of final layers, without accessing the labels.

Lee et al. propose the CleanNet [50], which uses a pre-defined subset of reference images. The visual features of the reference subset are extracted using autoencoder and each new sample for training is compared with the features from the reference set. Based on the distance, a weight is set for each sample and the weighted cross-entropy is calculated.

D. Sample Selection

Jiang et al. [16] proposes MentorNet, which uses a curriculum scheme by learning first the samples, which are probably correct. MentorNet learns a data-driven curriculum dynamically with a second network, called StudentNet.

Co-teaching is proposed by Han et al. [51] and trains two deep models simultaneously, and each network selects the batch of data for each other, based on the samples with a small loss. Later, they propose Co-teaching+ [17], that uses the samples with a small loss, but that disagree on the predictions, to select the data to each other. Wei et al. [52] uses the same idea of CoTeaching, but it uses the joint agreements instead of disagreement and calculates a joint loss with Co-Regularization.

Nguyen et al. propose an algorithm called SELF [53]. SELF uses a filtering mechanism based on a model ensemble learning the remove the noisy samples from the supervised training. However, they still use the removed samples to leverage the learning using an unsupervised loss. The ensemble mechanism is based on the model's predictions in different epochs, using an exponential moving average of model snapshots. Different from most of the state of the art methods, SELF requires a small clean validation set to perform well.

E. Meta Learning

The methods described here use as main approach Meta-Learning models to deal with noisy labels. Although in the end they use meta-learning to reweight or filter samples, we grouped them here because they use a different approach to address the problem.

Ren et al. [24] use a Meta-Learning paradigm to reweight the training samples based on their gradients directions. They perform a meta gradient descent step to minimize the loss on

a clean validation set. In [54] it is also used a meta-learning approach with a clean validation set, but they use a multi-layer perceptron to learn a loss-weighting function.

Li et al. [19] propose optimize a meta-objective before conventional training. By generating synthetic noisy labels, they aim to optimize a model that does not overfit to a wide spectrum of artificially generated label noise. By training the model in several generated synthetic noisy, they aim to have a noise-tolerant model able to consistently learn the underlying knowledge from data despite different label noise.

F. Combined Approaches

Mixup [55] is a technique proposed for data augmentation, that uses a linear combination between samples and labels. Their paper shows that their method is noise-tolerant, and recent literature methods started to incorporate Mixup as an important part of their algorithms. Zhang et al. [56] combine the reweighting approach using meta-learning, proposed by [24], with pseudo-label estimation and Mixup. Although they achieve state of the art for high noise regimes, it requires a small clean validation set.

Kim et al. [27] propose the use of the learning method called Negative Learning (NL), where it is used as a complementary label (i.e., a label that is different from the annotation). By using complementary labels, the chances of selecting a true label as a complementary are low, and NL decreases the risk of providing incorrect information. Therefore, training with NL is more robust to noise. However, this approach requires a longer training time, and the chance of providing incorrect information increases as the number of classes in the problem increases. In their approach, they propose three stages: Negative Learning, Positive Learning (standard training), and fine-tuning with relabeled samples.

Han et al. [57] propose to estimate class prototypes using an iterative training divided into two stages. The first stage train the network with the original noisy labels and the modified labels from the second stage. The second stage uses the trained network from the first stage and refines the prototypes, relabelling samples. With the automatic identification of prototypes, it does not need the use of a clean auxiliary set.

The DivideMix approach [58] combines several methods used in literature to deal with the noisy label problem. It first separates the samples in clean and noise based on Arazo's approach [18], using Mixture of Gaussians. At the same time, it uses the co-training strategy, where it trains two networks at the same time to avoid error accumulation. After it splits the data in clean and noisy, it trains the model in a semi-supervised approach, using MixMatch [59]. The whole process is repeated at each epoch.

The main approaches described in this section are summarized in Table I that shows the combined strategies used by the methods of the state-of-the-art. Table II shows the main components of the combined techniques in the state-of-the-art and the difference between them.

IV. Experimental Setup

Most of the papers in the literature evaluate their methods by generating synthetic noise on commonly used data sets and by using data set with images collected from the internet, which may contain closed-set and open-set noise. The most used data sets used for robust model evaluations in literature are CIFAR-10/CIFAR-100 [30], Clothing1M [60], Webvision [61] and Food101-N [50].

CIFAR-10 has 10 classes with 5000 32×32 pixel training images per class (forming a total of 50000 training images), and a testing set with 10000 32×32 pixel images with 1000 images per class. CIFAR-100 has 50000 training images, but with 100 classes with 5000 32×32 pixel images per class. This data set was curated and it is assumed not to have any noisy label. Therefore, it is a common approach to add synthetic noise and evaluate the robustness of the model in a noisy data set compared to the original clean version.

Clothing1M consists of 1 million training images acquired from online shopping websites, with labels generated by surrounding texts provided by sellers. The images from the data set may vary in size, but a common approach is to resize the images to 256×256 for training. This data set contains real-world error on the labels and it is composed of 14 classes.

The Webvision contains 2.4 million images collected from the internet, with the same classes from ILSVRC12, from ImageNet [7]. The images from the data set are not all with the same size, being a common approach to resize the images to 256×256 for training. Although ImageNet has 1000 classes, most of the papers use only the 50 first classes for training, because they contain most part of the the noise. The evaluation from the models trained using Webvision is usually done using both Webvision and ILSVRC12 validation sets.

Food101-N [50] is an image data set containing about 310009 training images of food recipes classified in 101 classes and 25000 images for the testing set. As the image size vary may vary, a common approach to resize the images to 256×256 for training. Its also a real-world data set, with an estimated noise rate of 20%, and it is based on the Food101 data set [62], but it has more images and it is more noisy.

Table IV shows the main information about the most used data sets for evaluation of solutions in noisy label environments.

V. State-of-the-art Results

Most of the state-of-the-art methods use the fact that CNN tends to learn easy patterns first and then fit the hardest ones [63]. Arazo et al. [18] showed in his paper how the values of loss are different among clean and noisy samples. Figure 3 shows the behavior of loss function for clean and noisy, reported in [18]. The strategy of filtering the samples based on loss is called *small trick* and has been exploited to identify clean and noisy samples. However, the main problem is that hard samples with correct labels can behave like noisy samples, and at the same time, some noisy samples can behave like a clean sample. Therefore, this approach can not be used to filter all samples, but it helps identify most of the noise. Arazo

TABLE I
MAIN APPROACHES IN THE LITERATURE TO DEAL WITH NOISY LABELS.

Approaches	Methods	Advantages	Disadvantages
Transition Matrix	[29], [33]–[39]	Easy to implement.	Difficult and complex to estimate the transition matrix in practice.
Robust Losses	[20], [21], [31], [40]–[45]	It can be easily added to any training model.	Requires to be combined with other strategies to be competitive with state-of-the-art.
Sample Weighting	[46]–[50]	Reduces the influence of noisy samples, but still uses information from it.	Hard to define a correct weighting without the need of a clean auxiliary set.
Sample Selection	[16], [17], [51]–[53]	Filter clean samples.	Not competitive with state-of-the-art because do not use the noisy samples in an unsupervised way.
Meta-Learning	[19], [24], [54]	Has big potential of generalization among different tasks.	Usually requires a clean validation set.
Combined	[18], [24], [27], [55], [56], [58]	Good performance, being the state-of-the-art.	Use a set of combined methods that adds complexity to the solution.

TABLE II
STATE-OF-THE-ART APPROACHES AND COMBINED TECHNIQUES

Method	Mixup (Data aug.)	Weighting	filtering	robust loss	regularization	Emsemble	Pseudo-labeling	val. set
NLNL [27]				✓	✓		✓	
SELF [53]			✓	✓			✓	✓
M-correction [18]			✓		✓			
Zhang [56]	✓	✓			✓			✓
Coteaching+ [17]			✓			✓		
DivideMix [58]	✓		✓		✓	✓		

TABLE III
MAIN DATA SETS USED IN LITERATURE FOR NOISY LABELS.

Data set	# of training	# of testing	# of class	estimated noise
CIFAR-10 [30]	50000	10000	10	0%
CIFAR-100 [30]	50000	10000	100	0%
Clothing1M [60]	1M	10000	14	38.46%
Webvision [61]	1M	50000	1000	20%
Food101-N [50]	310000	55000	101	19.66%

et al. proposes the use of a Gaussian Mixture Model (GMM) to separate the clean and noise during the training, based on the loss value of each sample. For the samples predicted as clean, it uses standard training, with regular cross-entropy, while the the samples predict as noisy are used to regularize the model in an unsupervised way.

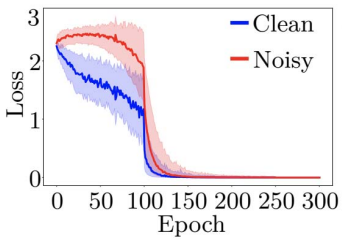
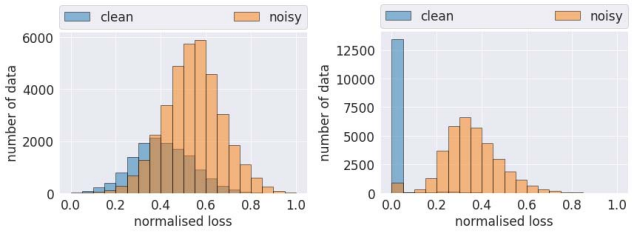


Fig. 3. Cross-Entropy loss of clean and noisy samples, for 80% noisy rate, for CIFAR-10. Figure from Arazo's paper [18].

DivideMix achieves a better split of clean and noisy samples by using the *small trick* combined with Mixup algorithm.



(a) Epoch 1 (b) Epoch 300

Fig. 4. Loss histogram for clean and noisy samples, using DivideMix, for 80% noise rate, symmetric noise, for CIFAR-10.

Figure 4 shows the results of the split for 80% symmetric noise rate for CIFAR-10.

DivideMix is currently the state-of-the-art for noisy labels, considering symmetric and asymmetric closed-set noise, without requiring a clean validation set. As described in section III, DivideMix is a combination of methods, which combines Arazo's approach, using GMM to separate clean and noisy samples, Mixup data augmentation and co-training strategy. It also uses standard data augmentation, such as rotation and flipping, as most of the methods described. The main idea of DivideMix is to separate the clean and noisy samples, using GMM, and then treat the problem as a semi-supervised problem, using MixMatch algorithm, that is a variation of the Mixup method proposed for semi-supervised problem. Furthermore, the co-training strategy helps with the noisy training.

As different methods in literature use different model archi-

tectures and sometimes different data sets, it is hard to make a fair comparison between most of them. In Table IV we show the state-of-the-art for CIFAR-10, CIFAR-100, Webvision and Clothing1M, using PreActResNet18 (PRN18). We did not include the methods which use an auxiliar clean validation set, such as in [53] and [56], to make a fair comparison, and only the methods that use PRN18 in the original paper. Table VI shows the SOTA results for Clothing1M. Table V shows the SOTA results for Webvision and ImageNet. Table VII shows the SOTA results for Food-101N. All the results are the ones reported in the original papers. The original code for reproduction of the results are also available and reported in original papers.

TABLE IV
SOTA RESULTS FOR CIFAR-10 AND CIFAR-100, USING PRN18. COMPARISON RESULTS ADAPTED FROM [58]

Data set	CIFAR-10					CIFAR-100			
Noise type	sym.				asym.	sym.			
Method/ noise ratio	20%	50%	80%	90%	40%	20%	50%	80%	90%
Cross-Entropy [58]	86.8	79.4	62.9	42.7	85.0	62.0	46.7	19.9	10.1
Coteaching+ [17]	89.5	85.7	67.4	47.9	-	65.6	51.8	27.9	13.7
Mixup [55]	95.6	87.1	71.6	52.2	-	67.8	57.3	30.8	14.6
PENCIL [64]	92.4	89.1	77.5	58.9	88.5	69.4	57.5	31.1	15.3
Meta-Learning [19]	92.9	89.3	77.4	58.7	89.2	68.5	59.2	42.4	19.5
M-correction [18]	94.0	92.0	86.8	69.1	87.4	73.9	66.1	48.2	24.3
DivideMix [58]	**96.1**	**94.6**	**93.2**	**76.0**	**93.4**	**77.3**	**74.6**	**60.2**	**31.5**

TABLE V
SOTA RESULTS FOR WEBVISION AND IMAGENET ILSVRC12. RESULTS ARE ADAPTED FROM [58].

Method	Webvision	ILSVRC12
F-correction [29]	61.12	57.36
MentorNet [16]	63.00	57.80
Co-teaching [51]	63.58	61.48
Iterative-CV [65]	65.24	61.60
DivideMix [58]	**77.32**	**75.20**

TABLE VI
SOTA RESULTS FOR CLOTHING1M. RESULTS ARE ADAPTED FROM [58].

Method	Test Accuracy
Cross-Entropy [58]	69.21
M-correction [18]	71.00
PENCIL [64]	73.49
DeepSelf [57]	74.45
CleanNet [50]	74.69
DivideMix [58]	**74.76**

TABLE VII
SOTA RESULTS FOR FOOD-101N.

Method	Food101-N
Cross-Entropy [50]	81.44
CleanNet [50]	83.95
DeepSelf [57]	**85.11**

VI. CONCLUSION

Several studies have been proposed in the literature to address the noise label problem. Different strategies have been investigated to make the training of deep learning models more robust to noise labels. Combined strategies based on data augmentation, robust loss, sample filtering, and semi-supervised approaches are currently state-of-the-art. Although we have seen an increasing interest in noisy label problems, there is still much room for improvement, mainly related to asymmetric noise and open-set noise.

Addressing the noisy label problem also impacts other areas, such as pseudo-labeling, semi-supervised and unsupervised training, where recent proposals use predicted labels to improve the training, and these labels can potentially be incorrect. At the same time, many recent strategies from these fields are also applied to noisy labels.

Recent advances have shown that the loss values of samples, mainly at the beginning of training, can help separate the clean and noisy samples. Moreover, data augmentation strategies, such as Mixup, can prevent the model from easily memorizing the noisy samples. However, it is an open question of how to differentiate hard clean samples from noisy samples. Also, semantic noise and open-set noise must be more investigated in future works.

REFERENCES

[1] W. Rawat and Z. Wang, "Deep convolutional neural networks for image classification: A comprehensive review," *Neural computation*, vol. 29, no. 9, pp. 2352–2449, 2017.

[2] F. Lateef and Y. Ruichek, "Survey on semantic segmentation using deep learning techniques," *Neurocomputing*, vol. 338, pp. 321–348, 2019.

[3] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *International journal of computer vision*, vol. 128, no. 2, pp. 261–318, 2020.

[4] T. J. Brinker, A. Hekler, A. H. Enk, C. Berking, S. Haferkamp, A. Hauschild, M. Weichenthal, J. Klode, D. Schadendorf, T. Holland-Letz *et al.*, "Deep neural networks are superior to dermatologists in melanoma image classification," *European Journal of Cancer*, vol. 119, pp. 11–17, 2019.

[5] Y. Li, C. Huang, L. Ding, Z. Li, Y. Pan, and X. Gao, "Deep learning in bioinformatics: Introduction, application, and perspective in the big data era," *Methods*, vol. 166, pp. 4–21, 2019.

[6] S. Mahdavifar and A. A. Ghorbani, "Application of deep learning to cybersecurity: A survey," *Neurocomputing*, vol. 347, pp. 149–176, 2019.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[8] X. Xie, J. Mao, Y. Liu, M. de Rijke, Q. Ai, Y. Huang, M. Zhang, and S. Ma, "Improving web image search with contextual information," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2019, pp. 1683–1692.

[9] X. Yu, T. Liu, M. Gong, and D. Tao, "Learning with biased complementary labels," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 68–83.

[10] E. Barkan, A. Hazan, and V. Ratner, "Reduce discrepancy of human annotators in medical imaging by automatic visual comparison to similar cases," Oct. 31 2019, uS Patent App. 15/963,120.

[11] K. Ma, X. Liu, Y. Fang, and E. P. Simoncelli, "Blind image quality assessment by learning from multiple annotators," in *2019 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2019, pp. 2344–2348.

[12] D. McNicol, *A primer of signal detection theory*. Psychology Press, 2005.

[13] B. Frénay and M. Verleysen, "Classification in the presence of label noise: a survey," *IEEE transactions on neural networks and learning systems*, vol. 25, no. 5, pp. 845–869, 2013.

[14] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," 2018.

[15] X. Wang, Y. Hua, E. Kodirov, and N. M. Robertson, "Proselflc: Progressive self label correction for training robust deep neural networks," *arXiv preprint arXiv:2005.03788*, 2020.

[16] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, and L. Fei-Fei, "Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels," in *International Conference on Machine Learning*, 2018, pp. 2304–2313.

[17] X. Yu, B. Han, J. Yao, G. Niu, I. W. Tsang, and M. Sugiyama, "How does disagreement help generalization against label corruption?" *arXiv preprint arXiv:1901.04215*, 2019.

[18] E. Arazo, D. Ortego, P. Albert, N. O'Connor, and K. Mcguinness, "Unsupervised label noise modeling and loss correction," in *International Conference on Machine Learning*, 2019, pp. 312–321.

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

[20] X. Wang, Y. Hua, E. Kodirov, and N. M. Robertson, "Imae for noise-robust learning: Mean absolute error does not treat examples equally and gradient magnitude's variance matters," *arXiv preprint arXiv:1903.12141*, 2019.

[21] Y. Wang, X. Ma, Z. Chen, Y. Luo, J. Yi, and J. Bailey, "Symmetric cross entropy for robust learning with noisy labels," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 322–330.

[22] L. Jaehwan, Y. Donggeun, and K. Hyo-Eun, "Photometric transformer networks and label adjustment for breast density prediction," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019, pp. 0–0.

[23] B. Yuan, J. Chen, W. Zhang, H.-S. Tai, and S. McMains, "Iterative cross learning on noisy labels," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 2018, pp. 757–765.

[24] M. Ren, W. Zeng, B. Yang, and R. Urtasun, "Learning to reweight examples for robust deep learning," in *International Conference on Machine Learning*, 2018, pp. 4334–4343.

[25] B. Han, G. Niu, J. Yao, X. Yu, M. Xu, I. Tsang, and M. Sugiyama, "Pumpout: A meta approach for robustly training deep neural networks with noisy labels," 2018.

[26] Q. Miao, Y. Cao, G. Xia, M. Gong, J. Liu, and J. Song, "Rboost: Label noise-robust boosting algorithm based on a nonconvex loss function and the numerically stable base learners," *IEEE transactions on neural networks and learning systems*, vol. 27, no. 11, pp. 2216–2228, 2015.

[27] Y. Kim, J. Yim, J. Yun, and J. Kim, "Nlnl: Negative learning for noisy labels," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 101–110.

[28] W. Zhang, Y. Wang, and Y. Qiao, "Metacleaner: Learning to hallucinate clean representations for noisy-labeled visual recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7373–7382.

[29] G. Patrini, A. Rozza, A. Krishna Menon, R. Nock, and L. Qu, "Making deep neural networks robust to label noise: A loss correction approach," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1944–1952.

[30] A. Krizhevsky, G. Hinton *et al.*, "Learning multiple layers of features from tiny images," 2009.

[31] Z. Zhang and M. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *Advances in neural information processing systems*, 2018, pp. 8778–8788.

[32] D. Angluin and P. Laird, "Learning from noisy examples," *Machine Learning*, vol. 2, no. 4, pp. 343–370, 1988.

[33] D. Hendrycks, M. Mazeika, D. Wilson, and K. Gimpel, "Using trusted data to train deep networks on labels corrupted by severe noise," in *Advances in neural information processing systems*, 2018, pp. 10 456–10 465.

[34] S. Sukhbaatar and R. Fergus, "Learning from noisy labels with deep neural networks," *arXiv preprint arXiv:1406.2080*, vol. 2, no. 3, p. 4, 2014.

[35] S. E. Reed, H. Lee, D. Anguelov, C. Szegedy, D. Erhan, and A. Rabinovich, "Training deep neural networks on noisy labels with bootstrapping," *arXiv preprint arXiv:1412.6596*, 2014.

[36] J. Goldberger and E. Ben-Reuven, "Training deep neural-networks using a noise adaptation layer," *ICLR*, 2017.

[37] X. Chen and A. Gupta, "Webly supervised learning of convolutional networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1431–1439.

[38] A. J. Bekker and J. Goldberger, "Training deep neural-networks based on unreliable labels," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 2682–2686.

[39] X. Xia, T. Liu, N. Wang, B. Han, C. Gong, G. Niu, and M. Sugiyama, "Are anchor points really indispensable in label-noise learning?" in *Advances in Neural Information Processing Systems*, 2019, pp. 6838–6849.

[40] N. Manwani and P. Sastry, "Noise tolerance under risk minimization," *IEEE transactions on cybernetics*, vol. 43, no. 3, pp. 1146–1151, 2013.

[41] A. Ghosh, H. Kumar, and P. Sastry, "Robust loss functions under label noise for deep neural networks," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[42] L. Ziyin, B. Chen, R. Wang, P. P. Liang, R. Salakhutdinov, L.-P. Morency, and M. Ueda, "Learning not to learn in the presence of noisy labels," *arXiv preprint arXiv:2002.06541*, 2020.

[43] S. Thulasidasan, T. Bhattacharya, J. Bilmes, G. Chennupati, and J. Mohd-Yusof, "Combating label noise in deep learning using abstention," in *International Conference on Machine Learning*, 2019, pp. 6234–6243.

[44] X. Ma, H. Huang, Y. Wang, S. Romano, S. Erfani, and J. Bailey, "Normalized loss functions for deep learning with noisy labels," *ICML*, 2020.

[45] Y. Liu and H. Guo, "Peer loss functions: Learning from noisy labels without knowing noise rates," *ICML*, 2019.

[46] Y. Wang, W. Liu, X. Ma, J. Bailey, H. Zha, L. Song, and S.-T. Xia, "Iterative learning with open-set noisy labels," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8688–8696.

[47] S. Guo, W. Huang, H. Zhang, C. Zhuang, D. Dong, M. R. Scott, and D. Huang, "Curriculumnet: Weakly supervised learning from large-scale web images," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 135–150.

[48] C. Xue, Q. Dou, X. Shi, H. Chen, and P.-A. Heng, "Robust learning at noisy labeled medical images: Applied to skin lesion classification," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*. IEEE, 2019, pp. 1280–1283.

[49] H. Harutyunyan, K. Reing, G. Ver Steeg, and A. Galstyan, "Improving generalization by controlling label-noise information in neural network weights," *ICML*, 2020.

[50] K.-H. Lee, X. He, L. Zhang, and L. Yang, "Cleannet: Transfer learning for scalable image classifier training with label noise," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5447–5456.

[51] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," in *Advances in neural information processing systems*, 2018, pp. 8527–8537.

[52] H. Wei, L. Feng, X. Chen, and B. An, "Combating noisy labels by agreement: A joint training method with co-regularization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 13 726–13 735.

[53] T. Nguyen, C. Mummadi, T. Ngo, L. Beggel, and T. Brox, "Self: learning to filter noisy labels with self-ensembling," in *International Conference on Learning Representations (ICLR)*, 2020.

[54] J. Shu, Q. Xie, L. Yi, Q. Zhao, S. Zhou, Z. Xu, and D. Meng, "Meta-weight-net: Learning an explicit mapping for sample weighting," in *Advances in Neural Information Processing Systems*, 2019, pp. 1919–1930.

[55] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *International Conference on Learning Representations*, 2018.

[56] Z. Zhang, H. Zhang, S. O. Arik, H. Lee, and T. Pfister, "Distilling effective supervision from severe label noise," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9294–9303.

[57] J. Han, P. Luo, and X. Wang, "Deep self-learning from noisy labels," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 5138–5147.

[58] J. Li, R. Socher, and S. C. Hoi, "Dividemix: Learning with noisy labels as semi-supervised learning," *arXiv preprint arXiv:2002.07394*, 2020.
[59] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "Mixmatch: A holistic approach to semi-supervised learning," in *Advances in Neural Information Processing Systems*, 2019, pp. 5049–5059.
[60] T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang, "Learning from massive noisy labeled data for image classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 2691–2699.
[61] W. Li, L. Wang, W. Li, E. Agustsson, and L. V. Gool, "Webvision database: Visual learning and understanding from web data." *CoRR*, 2017.
[62] L. Bossard, M. Guillaumin, and L. Van Gool, "Food-101–mining discriminative components with random forests," in *European conference on computer vision*. Springer, 2014, pp. 446–461.
[63] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," *International Conference On Learning Representations (ICLR)*, vol. abs/1611.03530, 2017.
[64] K. Yi and J. Wu, "Probabilistic end-to-end noise correction for learning with noisy labels," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7017–7025.
[65] P. Chen, B. Liao, G. Chen, and S. Zhang, "Understanding and utilizing deep neural networks trained with noisy labels," *arXiv preprint arXiv:1905.05040*, 2019.