Pattern Recognition 134 (2023) 109121

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

ScanMix: Learning from Severe Label Noise via Semantic Clustering and Semi-Supervised Learning

Ragav Sachdeva[a,1,*], Filipe Rolim Cordeiro[c], Vasileios Belagiannis[d], Ian Reid[b], Gustavo Carneiro[b,e]

[a] Visual Geometry Group, Department of Engineering Science, University of Oxford, United Kingdom
[b] School of Computer Science, Australian Institute for Machine Learning, Australia
[c] Universidade Federal Rural de Pernambuco, Brazil
[d] Otto-von-Guericke-Universität Magdeburg, Germany
[e] Centre for Vision, Speech and Signal Processing, University of Surrey, United Kingdom

ARTICLE INFO

Article history:
Received 10 December 2021
Revised 17 August 2022
Accepted 16 October 2022
Available online 19 October 2022

Keywords:
Noisy label learning
Semi-supervised learning
Semantic clustering
Self-supervised Learning
Expectation maximisation

ABSTRACT

We propose a new training algorithm, ScanMix, that explores semantic clustering and semi-supervised learning (SSL) to allow superior robustness to severe label noise and competitive robustness to non-severe label noise problems, in comparison to the state of the art (SOTA) methods. ScanMix is based on the expectation maximisation framework, where the E-step estimates the latent variable to cluster the training images based on their appearance and classification results, and the M-step optimises the SSL classification and learns effective feature representations via semantic clustering. We present a theoretical result that shows the correctness and convergence of ScanMix, and an empirical result that shows that ScanMix has SOTA results on CIFAR-10/-100 (with symmetric, asymmetric and semantic label noise), Red Mini-ImageNet (from the Controlled Noisy Web Labels), Clothing1M and WebVision. In all benchmarks with severe label noise, our results are competitive to the current SOTA.

1. Introduction

Much of the success of deep learning models is attributable to the availability of well-curated large-scale datasets that enables a reliable supervised learning process [1]. However, the vast majority of real-world datasets have noisy labels due to human failure, poor quality of data or inadequate labelling process [2]. Using noisy label datasets for training not only hurts the model's accuracy, but also biases the model to make the same mistakes present in the labels [3]. Therefore, one of the important challenges in the field is the formulation of robust training algorithms that work effectively with datasets corrupted with noisy labels.

Successful approaches to address the learning from noisy label (LNL) problem tend to rely on semi-supervised learning (SSL) [4–7]. Such methods run the following steps iteratively: a) automatically split the training set into clean and noisy sets, b) discard the labels of the samples in the noisy set and, c) minimise the classification loss with the labelled (clean) and unlabelled (noisy) data. Consequently, these SSL methods rely on successfully splitting the training set into clean and noisy sets, which for low noise rates, is accurate [4–7] because of the strong support in the training set that associates image representations and their true labels. However, for severe label noise, this support weakens, resulting in the over-fitting of label noise [4–7].

To mitigate the issues caused by severe label noise, one can consider self-supervised learning strategies [8–11] to build feature representations using appearance clustering techniques. These strategies [8–11] show better classification accuracy than recently proposed LNL methods [7,12] when the noise rate is large (above 80% symmetric and 40% asymmetric). However, for low noise rates, SSL methods tend to produce better results because self-supervised methods typically tend to cluster images with similar appearance, but such similarity does not imply that the images belong to same class. We argue that in a noisy label context, the use of self-supervised learning (without using the training set labels) *can* create an initial feature representation that is more related to the real hidden representation *in comparison to* supervised training with noisy labels. However, when the dataset is relatively well-structured and clean, the use of self-supervised learning alone is

* Corresponding author.
E-mail address: removethisifyouarehuman-rs@robots.ox.ac.uk (R. Sachdeva).
[1] *Present Address:* Visual Geometry Group, Department of Engineering Science, University of Oxford, United Kingdom

https://doi.org/10.1016/j.patcog.2022.109121

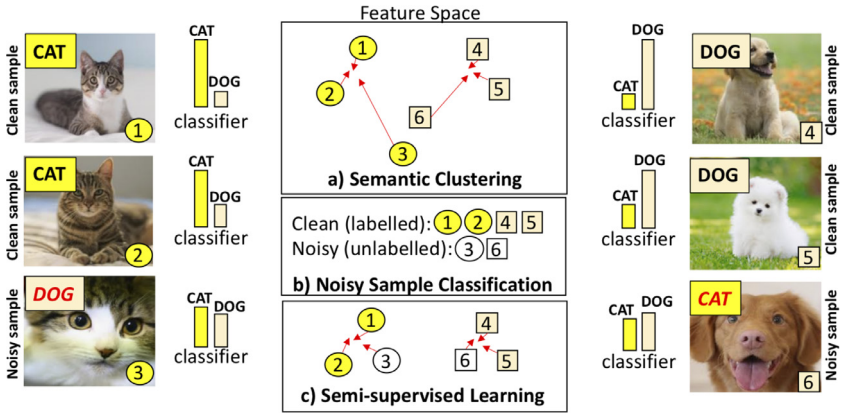

Fig. 1. ScanMix explores **semantic clustering** (a) that clusters samples with similar appearances and classification results, and **semi-supervised learning (SSL)** (c) that trains the classifier by treating the samples classified to have noisy labels, as unlabelled samples. In the figure, circles represent true cat label, and squares, true dog class, where samples 3 and 6 are noisy, but the classifier produces the right classification (see yellow and pink bars). In frames (a),(c) the arrows denote how the training process moves samples in the feature space at each stage, with samples 3 and 6 showing white background in (b),(c) because they are classified as noisy in (b) and have their labels removed for SSL.

not enough to bridge the gap to its supervised training counterpart. To this end, we propose a training mechanism that performs semantic clustering and SSL in tandem. We hypothesise that such training 1) enables the model to not get too biased by noisy labels (as it is guided by semantic clustering), and 2) still produces accurate classification results using the SSL strategy. These points are illustrated in Fig. 1.

We test this hypothesis with the proposed label-noise robust training algorithm, ScanMix, that explores semantic clustering and SSL to enable superior robustness to severe label noise and competitive robustness to non-severe label noise problems, compared with the state of the art (SOTA). ScanMix is based on the expectation maximisation (EM) framework [13], where the semantic clustering stage clusters images with similar appearance *and* classification results, enabling a more effective identification of noisy label images to subsequently be "unlabelled" and used by SSL. Although the use of EM in the context of LNL has been explored in [14], ScanMix is the first to propose the joint exploration of semantic clustering and SSL together. The implementation of ScanMix relies on SOTA semantic clustering [10] and noisy label robust SSL [7]. We show a theoretical result that proves that ScanMix is correct and converges to a stationary point under certain conditions. The main contributions of ScanMix are:

- A new noisy-label learning algorithm, based on EM optimisation, that explores and combines the advantages of semantic clustering and semi-supervised learning showing remarkable robustness to severe label noise rates;
- A new theoretical result that shows the correctness and convergence of the noisy-label learning algorithm; and
- Competitive performance in a wide range of noisy-label learning problems, such as symmetric, asymmetric, semantic or instance dependent, and controlled noisy web labels.

Empirical results on CIFAR-10/-100 [15] under symmetric, asymmetric and semantic noise, show that ScanMix outperforms previous approaches. For high-noise rate problems in CIFAR-10/-100 [7] and Red Mini-ImageNet from the Controlled Noisy Web Labels [16], ScanMix presents the best results in the field. Furthermore, we show results on the challenging semantic label noise present in the large-scale real-world datasets Clothing1M [17] and WebVision [18], where our proposed method shows SOTA results.

2. Prior Work

The main noisy label learning techniques are: label cleansing [19,20], iterative label correction [21], robust loss functions [22–24], meta-learning [16,25,26], sample weighting [27], ensemble learning [28], student-teacher model [29], co-teaching [7,30–33], dimensionality reduction of the image representation [34], and combinations of the techniques above [12,35–38]. Recent advances showed that the most promising strategies are based on the combination of co-training, noise filtering, data augmentation and SSL [5–7]. Below, we do not review approaches that require a clean validation set, such as [39], since that setup imposes a strong constraint on the type of noisy-label learning problem.

Instead, we focus on methods based on SSL for noisy-label training [4–7,40,41], which usually show SOTA results on several benchmarks. These methods rely on: 1) the identification of training samples containing noisy labels and the subsequent removal of their labels; and 2) performing SSL [42] using this set of unlabelled samples and the remaining set of labelled samples. [4] identify a small portion of clean samples from the noisy training set by associating them with high confidence. Then, they use the filtered samples as labelled and the remaining ones as unlabelled in an SSL approach. However, relying on highly confident samples to compose the labelled set may not work well for severe noise rate scenarios because this labelled set can be contaminated with high noise rate. The methods in [5,7] classify the noisy and clean samples by fitting a two-component Gaussian Mixture Model (GMM) on the normalised loss values for each training epoch. Next, they use MixMatch [42] to combine the labelled and unlabelled sets with MixUp [43]. However, these strategies do not perform well for high noise rates because MixUp tends to be ineffective in such scenario. SSL methods can be robust to severe label noise by exploring a feature clustering scheme that pulls together samples that are semantically similar, without considering the noisy labels from the training set, and one way to enable such semantic clustering is provided by self-supervised learning [8–11].

Self-supervised learning has been used as a pre-training approach to estimate reliable features from unlabelled datasets, but we are not aware of methods that use it for semantic clustering. For instance, SimCLR [11] generates data augmentations of the input images and trains the model to have similar representation of an image and its augmented samples, while increasing the dissimilarity to the other images. MoCo [8,9] tackles self-supervised rep-

resentation learning by conflating contrastive learning with a dictionary look-up. The proposed framework builds a dynamic dictionary with a queue and a moving-averaged encoder to enable building a large and consistent dictionary on-the-fly that facilitates contrastive unsupervised learning. Another example is SCAN [10] that has several stages of self-supervised training: one based on SimCLR [11], followed by another based on a nearest neighbor clustering scheme, and another based on self-labelling. Self-supervised learning approaches usually show results better than the noisy label SOTA methods for severe label noise problems (above 80% noise), but for relatively low label noise rates (below 50% noise), self-supervised learning tends to be worse. Therefore, the main question we address in this paper is how to explore the semantic clustering capability of self-supervised learning approaches together with SSL methods, to improve the current SOTA results in severe label noise problems, and maintain the SOTA results in low noise label scenarios. Even though sophisticated clustering approaches have been proposed in the field [44–47], we opted to use a simple Euclidean-distance based K-nearest neighbour clustering approach. Furthermore, semantic clustering has been explored in LNL problems [14,48], but without relying on SSL methods.

3. Method

3.1. Dataset and Label Noise Types

Let the training set be denoted by $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{|\mathcal{D}|}$, with $\mathbf{x}_i \in \mathcal{S} \subseteq \mathbb{R}^{H\times W}$ being the i^{th} image, and $\mathbf{y}_i \in \{0,1\}^{|\mathcal{Y}|}$ a one-hot vector of the noisy label, where $\mathcal{Y} = \{1, ..., |\mathcal{Y}|\}$ represents the set of labels, and $\sum_{c\in\mathcal{Y}} \mathbf{y}_i(c) = 1$. The latent true label of the i^{th} training instance is denoted by $\hat{\mathbf{y}}_i \in \mathcal{Y}$, where $\sum_{c\in\mathcal{Y}} \hat{\mathbf{y}}_i(c) = 1$. This latent true label is used by a noise process to produce $\mathbf{y}_i \sim p(\mathbf{y}|\mathbf{x}_i, \mathcal{Y}, \hat{\mathbf{y}}_i)$, with $p(\mathbf{y}(j)|\mathbf{x}_i, \mathcal{Y}, \hat{\mathbf{y}}_i(c)) = \eta_{jc}(\mathbf{x}_i)$, where $\eta_{jc}(\mathbf{x}_i) \in [0,1]$ and $\sum_{j\in\mathcal{Y}} \eta_{jc}(\mathbf{x}_i) = 1$.

The types of noises considered in this paper are: symmetric [36], asymmetric [49], semantic [50], and real-world noise [17,18,38]. The symmetric (or uniform) noise flips the latent true label $\hat{\mathbf{y}}_i \in \mathcal{Y}$ to any of the labels in $\mathcal{Y}$ (including the true label) with a fixed probability η, so $\eta_{jc}(\mathbf{x}_i) = \frac{\eta}{|\mathcal{Y}|-1}, \forall j, c \in \mathcal{Y}$, such that $j \neq c$, and $\eta_{cc}(\mathbf{x}_i) = 1-\eta$. The asymmetric noise flips the labels between semantically similar classes [49], so $\eta_{jc}(\mathbf{x}_i)$ is based on a transition matrix between classes $j, c \in \mathcal{Y}$, but not on $\mathbf{x}_i$. The semantic noise [50] also uses an estimated transition probability between classes $j, c \in \mathcal{Y}$ but takes into account the image $\mathbf{x}_i$ (i.e., it is an image conditional transition probability). Real-world noise [17,18,38] contains the noise types above in addition to the open-set noise, where the class $c \notin \mathcal{Y}$.

3.2. ScanMix

The proposed ScanMix training algorithm (Fig. 2) is formulated with an EM algorithm that uses a latent random variable $z_{ji} \in \{0,1\}$ which indicates if a sample $\mathbf{x}_j$ belongs to the set of K nearest neighbours (KNN) of $\mathbf{x}_i$, estimated with the Euclidean distance. The classifier trained by ScanMix is parameterised by $\theta = [\psi, \phi] \in \Theta$, and represented by

$$p_\theta(\mathbf{y}|\mathbf{x}) = p_\psi(\mathbf{y}|f_\phi(\mathbf{x})), \tag{1}$$

where $p_\psi(.) \in [0,1]^{|\mathcal{Y}|}$ produces a probability distribution over the classes in the classification space $\mathcal{Y}$ using the feature representation $f_\phi(\mathbf{x}) \in \mathbb{R}^d$ of the input image $\mathbf{x}$.

The optimal parameters for the classifier are estimated with maximum likelihood estimation (MLE):

$$\theta^* = \arg\max_\theta \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}_i,\mathbf{y}_i)\in\mathcal{D}} \log p_\theta(\mathbf{y}_i|\mathbf{x}_i), \tag{2}$$

where

$$\begin{aligned}
\log p_\theta(\mathbf{y}_i|\mathbf{x}_i) &= \mathbb{E}_{q(z)}\left[\log\left(p_\theta(\mathbf{y}_i|\mathbf{x}_i)\frac{q(z)}{q(z)}\right)\right] \\
&= \int q(z)\log\left(\frac{p_\theta(\mathbf{y}_i, z|\mathbf{x}_i)q(z)}{p_\theta(z|\mathbf{y}_i,\mathbf{x}_i)q(z)}\right)dz \\
&= \mathbb{E}_{q(z)}[\log(p_\theta(\mathbf{y}_i, z|\mathbf{x}_i))] - \mathbb{E}_{q(z)}[\log q(z)] \\
&\quad + KL[q(z)||p_\theta(z|\mathbf{y}_i,\mathbf{x}_i)] \\
&= \ell_{ELBO}(q,\theta) + KL[q(z)||p_\theta(z|\mathbf{y}_i,\mathbf{x}_i)].
\end{aligned} \tag{3}$$

In Eq. 3 above, we have:

$$\ell_{ELBO}(q,\theta) = \mathbb{E}_{q(z)}[\log p_\theta(\mathbf{y}, z|\mathbf{x})] - \mathbb{E}_{q(z)}[\log q(z)], \tag{4}$$

with $KL[\cdot]$ denoting the Kullback-Leibler divergence, and $q(z)$ representing the variational distribution that approximates $p_\theta(z|\mathbf{y},\mathbf{x})$, defined as

$$\begin{aligned}
&p_\theta(z_{ji}|\mathbf{x}_i,\mathbf{y}_i) \\
&= \begin{cases} (1-z_{ji}), & \text{if } \mathbf{y}_j \neq \mathbf{y}_i \\ (p_\theta(:|\mathbf{x}_j)^\top p_\theta(:|\mathbf{x}_i))^{z_{ji}}(1-p_\theta(:|\mathbf{x}_j)^\top p_\theta(:|\mathbf{x}_i))^{(1-z_{ji})}, & \text{if } \mathbf{y}_j = \mathbf{y}_i \end{cases}
\end{aligned} \tag{5}$$

where $p_\theta(:|\mathbf{x}) \in [0,1]^{|\mathcal{Y}|}$ is the probability classification for defined in Eq. 1. Hence, Eq. 5 defines the probability of $z_{ij} \in \{0,1\}$, denoting the probability of $\mathbf{x}_j$ to belong to the set of K nearest neighbours (KNN) of $\mathbf{x}_i$. In this definition, when their labels are different, or $\mathbf{y}_i \neq \mathbf{y}_j$, the probability of $z_{ij} = 1$ is 0 (and consequently, the probability of $z_{ij} = 0$ is 1). Also, when their labels are equal, or $\mathbf{y}_i = \mathbf{y}_j$, the probability of $z_{ij} = 1$ depends on the similarity of their classification probabilities denoted by $p_\theta(:|\mathbf{x}_j)^\top p_\theta(:|\mathbf{x}_i)$.

The maximisation of the log likelihood in Eq. 2 follows the EM algorithm [13] consisting of two steps. The E-step maximizes the lower bound of Eq. 3 by zeroing the KL divergence with $q(z_{ji}) = p_{\theta^{old}}(z_{ji}|\mathbf{y}_i,\mathbf{x}_i)$, where θ^{old} denotes the parameter from the previous EM iteration. Then the M-step maximises ℓ_{ELBO} in Eq. 4, which re-writes Eq. 2 as:

$$\begin{aligned}
\theta^* &= \arg\max_\theta \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}_i,\mathbf{y}_i)\in\mathcal{D}} \sum_{j=1}^{|\mathcal{D}|} \sum_{z_{ij}\in\{0,1\}} q(z_{ji}) \log p_\theta(z_{ji}, \mathbf{y}_i|\mathbf{x}_i) \\
&= \arg\max_\theta \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}_i,\mathbf{y}_i)\in\mathcal{D}} \sum_{j=1}^{|\mathcal{D}|} \sum_{z_{ij}\in\{0,1\}} q(z_{ji}) \log\left(p_\theta(z_{ji}|\mathbf{y}_i,\mathbf{x}_i)p_\theta(\mathbf{y}_i|\mathbf{x}_i)\right)
\end{aligned}$$

which by noting that

$$\sum_{(\mathbf{x}_i,\mathbf{y}_i)\in\mathcal{D}} \sum_{j=1}^{|\mathcal{D}|} \sum_{z_{ij}\in\{0,1\}} q(z_{ji}) \log p_\theta(\mathbf{y}_i|\mathbf{x}_i) = \sum_{(\mathbf{x}_i,\mathbf{y}_i)\in\mathcal{D}} \sum_{j=1}^{|\mathcal{D}|} \log p_\theta(\mathbf{y}_i|\mathbf{x}_i),$$

we have

$$\begin{aligned}
\theta^* = \arg\max_\theta \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}_i,\mathbf{y}_i)\in\mathcal{D}} \sum_{j=1}^{|\mathcal{D}|} \\
\left(\log p_\theta(\mathbf{y}_i|\mathbf{x}_i) + \sum_{z_{ij}\in\{0,1\}} q(z_{ji}) \log p_\theta(z_{ji}|\mathbf{y}_i,\mathbf{x}_i)\right),
\end{aligned} \tag{6}$$

where the term $\mathbb{E}_{q(z)}[\log(q(z))]$ is removed from ℓ_{ELBO} since it only depends on the parameter from the previous iteration, θ^{old}. Hence, Eq. 6 comprises two terms: 1) the **classification term** that maximises the likelihood of the label $\mathbf{y}_i$ for sample $\mathbf{x}_i$; and 2) the **semantic clustering term** that maximises the association between samples that are close in the feature and label spaces, according to $q(z_{ji})$ estimated from the E-step.

According to the Equations 5 and 6, the run-time complexity of ScanMix is quadratic in $|\mathcal{D}|$, making this algorithm impractical for large-scale problems. Therefore, we approximate both steps by running a self-supervised pre-training process [8–11] that forms an initial set of K nearest neighbours in the feature space $f_\phi(\mathbf{x})$

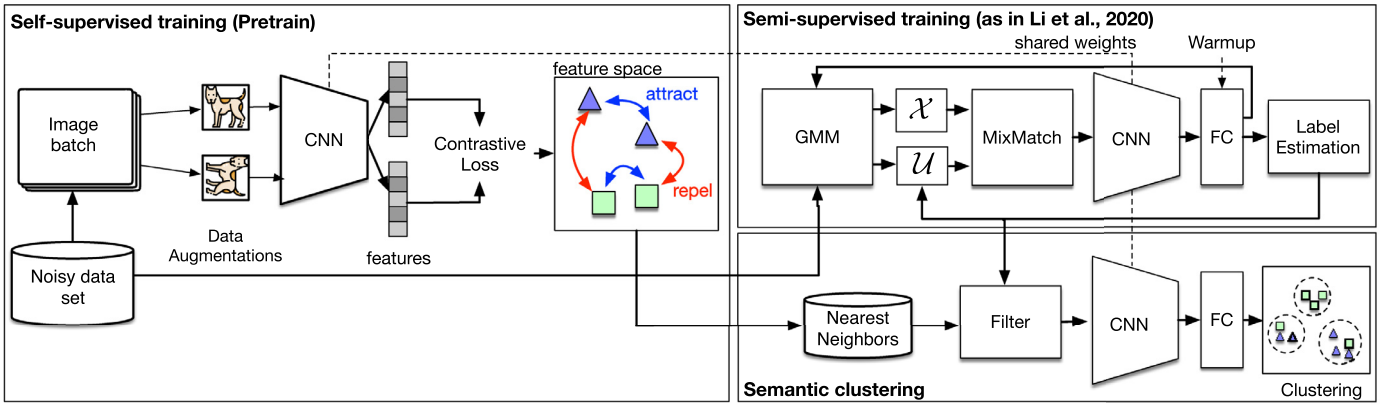

Fig. 2. ScanMix has a pre-training stage consisting of a self-supervised training [8–11], where we use contrastive loss to approximate features to its data augmented variants, in the feature space, while repelling representations from negative examples. In the training stage we first warm-up the classifier using a simple classification loss. Then, using the classification loss, we train the GMM to separate the samples into a clean set $\mathcal{X}$ and a noisy set $\mathcal{U}$ that are "MixMatched" [42] for SSL training. In parallel to this SSL training, we use the classification results and feature representations to train the semantic clustering. Please see Algorithm 1 for more details.

for each training sample. The set of KNN samples for each sample $\mathbf{x}_i \in \mathcal{D}$ is denoted by $\mathcal{N}_{\mathbf{x}_i} = \{\mathbf{x}_j\}_{j=1}^{K}$ (for $\mathbf{x}_j \in \mathcal{D}$). Then, $q(z_{ji})$ is approximated to be equal to 1, when $\mathbf{x}_j \in \mathcal{N}_{\mathbf{x}_i}$ and $\mathbf{y}_j = \mathbf{y}_i$, and 0 otherwise. Such an approximation makes the run-time complexity of the E and M steps linear in $\mathcal{D}$[2]. Also, using this approximation to estimate $q(z_{ji})$ in Eq. 5 reduces even more the complexity of the semantic clustering maximisation because we only consider $q(z_{ji}=1)$ instead of $q(z_{ji}=1)$ and $q(z_{ji}=0)$.

The optimisation of the **classification term** in Eq. 6 assumes that $\mathcal{D}$ is not noisy, so we modify it to enable learning with a noisy dataset. This is achieved by maximising a lower bound of that term, as follows [7]:

$$\begin{aligned} &\text{maximise} && \tfrac{1}{|\mathcal{X}|}\textstyle\sum_{(\mathbf{x},\mathbf{y})\in\mathcal{X}} \log p_\theta(\mathbf{y}|\mathbf{x}) \\ &\text{subject to} && \tfrac{1}{|\mathcal{U}|}\textstyle\sum_{(\mathbf{x},\mathbf{y})\in\mathcal{U}} \|\mathbf{y} - p_\theta(:|\mathbf{x})\|_2^2 = 0 \\ & && KL\left[\pi_{|\mathcal{Y}|} \middle|\middle| \tfrac{1}{|\mathcal{X}|+|\mathcal{U}|}\textstyle\sum_{\mathbf{x}\in(\mathcal{X}\cup\mathcal{U})} p_\theta(:|\mathbf{x})\right] = 0, \end{aligned} \tag{7}$$

where $\mathcal{X}$ and $\mathcal{U}$ represent the sets of samples extracted from $\mathcal{D}$ automatically classified as clean and noisy, respectively, $p_\theta(:|\mathbf{x})$ denotes the classification probability for all classes in $\mathcal{Y}$, $KL[.]$ represents the Kullback Leibler (KL) divergence, and $\pi_{|\mathcal{Y}|}$ denotes a vector of $|\mathcal{Y}|$ dimensions with values equal to $1/|\mathcal{Y}|$. The classification of training samples into clean or noisy is first formed with [7,38,50,51]:

$$\begin{aligned} \mathcal{X}' &= \{(\mathbf{x}_i, \mathbf{y}_i) : p(\text{clean}|\ell_i, \gamma) \geq \tau\}, \\ \mathcal{U}' &= \{(\mathbf{x}_i, \mathbf{y}_i^*) : p(\text{clean}|\ell_i, \gamma) < \tau\}, \end{aligned} \tag{8}$$

with τ denoting a threshold to classify a clean sample, $\mathbf{y}_i^* = p_\theta(:|\mathbf{x}_i)$, $\ell_i = -\mathbf{y}_i^\top \log p_\theta(:|\mathbf{x}_i)$, and $p(\text{clean}|\ell_i, \gamma)$ being a function that estimates the probability that $(\mathbf{x}_i, \mathbf{y}_i)$ is a clean label sample. The function $p(\text{clean}|\ell_i, \gamma)$ in Eq. 8 is a bi-modal Gaussian mixture model (GMM) [7] (γ denotes the GMM parameters), where the component with larger man is the noisy component and the smaller mean is the clean component. Next, we run SSL [7], consisting of a data augmentation to increase the number of samples in $\mathcal{X}'$ and $\mathcal{U}'$, followed by MixMatch [42] that combines samples from both sets to form the sets $\mathcal{X}$ and $\mathcal{U}$, which are used in Eq. 7. The optimisation in Eq. 7 is done with Lagrange multipliers by minimising the loss $\ell_{MLE} = \ell_{\mathcal{X}} + \lambda_u \ell_{\mathcal{U}} + \lambda_r \ell_r$, where $\ell_{\mathcal{X}}$ represents the (negative) objective function, $\ell_{\mathcal{U}}$ and ℓ_r denote the two constraints, and λ_u and λ_r are the Lagrange multipliers.

[2] We tested ScanMix without this approximation and preliminary results show that updating neighbors $\{N_{x_i}\}_{i=1}^{|\mathcal{D}|}$ leads to similar results as the ones in this paper, suggesting the validity of our approximation.

We constrain the optimisation of the **semantic clustering term** in Eq. 6 with a regulariser [10] to make it robust to semantic drift [52]. Hence, we maximise a lower bound of the semantic clustering term in Eq. 6, as follows:

$$\begin{aligned} &\text{maximise} \frac{1}{|\mathcal{D}|} \sum_{(\mathbf{x}_i,\mathbf{y}_i)\in\mathcal{D}} \sum_{j=1}^{|\mathcal{D}|} \sum_{z_{ij}\in\{0,1\}} q(z_{ji}) \log p_\theta(z_{ji}|\mathbf{y}_i, \mathbf{x}_i) \\ &\text{subject to} \sum_{c\in\mathcal{Y}} \mathbb{E}_{\mathbf{x}\sim\mathcal{D}}[p(c|\mathbf{x},\theta)] \log \mathbb{E}_{\mathbf{x}\sim\mathcal{D}}[p(c|\mathbf{x},\theta)] = 0, \end{aligned} \tag{9}$$

where $q(z_{ji}) = 1$ if $\mathbf{x}_i$ and $\mathbf{x}_j$ have the same classification result, i.e., $\arg\max_{c\in\mathcal{Y}} p_\theta(c|\mathbf{x}_i) = \arg\max_{c\in\mathcal{Y}} p_\theta(c|\mathbf{x}_j)$ and $\mathbf{x}_j \in \mathcal{N}_{\mathbf{x}_i}$. We also use Lagrange multipliers to optimise Eq. 9, where we minimise $\ell_{CLU} = \ell_{\mathcal{N}} + \lambda_e \ell_e$, with $\ell_{\mathcal{N}}$ denoting the negative objective function, ℓ_e representing the constraint, and λ_e being the Lagrange multiplier. An interesting point from the optimisations in Eq. 7 and Eq. 9 is that the constraints can help mitigate the semantic drift problem [52] typically present in under-constrained SSL methods.

3.3. Training, Inference, Correctness and Convergence Conditions

Algorithm 1 describes the training process that starts with a self-supervised pre-training [8–11] which optimises the parame-

Algorithm 1 ScanMix

Require: $\mathcal{D}$, number of epochs E, clean sample threshold τ

$f_\phi(\mathbf{x})$, $\{\mathcal{N}_{\mathbf{x}_i}\}_{i=1}^{|\mathcal{D}|}$ = PreTrain($\mathcal{D}$) ▷ Self-supervised pre-training
$p_\theta(\mathbf{y}|\mathbf{x})$ = WarmUp($\mathcal{D}$, $f_\phi(\mathbf{x})$) ▷ Warm Up
while $e < E$ **do**
  for $i = \{1, ..., |\mathcal{D}|\}$ **do**
    Estimate $p(\text{clean}|\ell_i, \gamma)$, with
    $\ell_i = -\mathbf{y}_i^\top \log p_\theta(:|\mathbf{x}_i)$
  $\mathcal{X}', \mathcal{U}'$=FormCleanNoisySets($\{p(\text{clean}|\ell_i, \gamma)\}_{i=1}^{|\mathcal{D}|}, \tau$)
  $\mathcal{X}, \mathcal{U}$=MixMatch($\mathcal{X}', \mathcal{U}'$)
  for $(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{D}$ **do** ▷ E-step
    $\tilde{y}_i = \arg\max_{c\in\mathcal{Y}} p_\theta(c|\mathbf{x}_i)$
    $q(z_{ji}) = 0, \forall j \in \{1, ..., |\mathcal{D}|\}$
    for $\mathbf{x}_j \in \mathcal{N}_{\mathbf{x}_i}$ **do**
      $\tilde{y}_j = \arg\max_{c\in\mathcal{Y}} p_\theta(c|\mathbf{x}_j)$
      if $(\tilde{y}_i == \tilde{y}_j)$ **then**
        $q(z_{ji}) = 1$
  Minimise ℓ_{MLE} with $\mathcal{X}$, $\mathcal{U}$, and
  ℓ_{CLU} with $\{\mathcal{N}_{\mathbf{x}_i}\}_{i=1}^{|\mathcal{D}|}$, and $\{q(z_{ji})\}_{i,j=1}^{|\mathcal{D}|}$ ▷ M-step

ters of the feature extractor $f_\phi(\mathbf{x})$ using the unlabelled images of $\mathcal{D}$ and defines the set of KNNs for each training sample $\{\mathcal{N}_{\mathbf{x}_i}\}_{i=1}^{|\mathcal{D}|}$. Then, we warm-up the classifier by training it for a few epochs on the (noisy) training dataset using cross-entropy loss. Next, we run the EM optimisation using Eq. 7 and Eq. 9. The inference uses the model in Eq. 1 to classify $\mathbf{x}$.

ScanMix improves $\ell_{ELBO}(q,\theta)$ in Eq. 4 instead of improving $p_\theta(\mathbf{y}|\mathbf{x})$ in Eq. 2. Following Theorem 1 in [13], Lemma 1 shows the correctness of ScanMix, where an improvement to $\ell_{ELBO}(q,\theta)$ implies an increase to $p_\theta(\mathbf{y}|\mathbf{x})$. Following Theorem 2 in [13], Lemma 2 shows the convergence conditions of ScanMix.

Lemma 1. *Assuming that the maximisation of ℓ_{ELBO} in Eq. 6 estimates θ that makes $\mathbb{E}_{q(z)}[\log p_\theta(\mathbf{y},z|\mathbf{x})] \geq \mathbb{E}_{q(z)}[\log p_{\theta^{old}}(\mathbf{y},z|\mathbf{x})]$, we have that $\left(\log p_\theta(\mathbf{y}|\mathbf{x}) - \log p_{\theta^{old}}(\mathbf{y}|\mathbf{x})\right)$ is lower bounded by $\left(\mathbb{E}_{q(z)}[\log p_\theta(\mathbf{y},z|\mathbf{x})] - \mathbb{E}_{q(z)}[\log p_{\theta^{old}}(\mathbf{y},z|\mathbf{x})]\right) \geq 0$, with $q(z) = p_{\theta^{old}}(z|\mathbf{y},\mathbf{x})$.*

Proof. Following the proof for Theorem 1 in [13], from Eq. 3, we have

$$\log p_\theta(\mathbf{y}|\mathbf{x}) \quad = \ell_{ELBO}(q,\theta) + KL[q(z)||p_\theta(z|\mathbf{y},\mathbf{x})], \tag{10}$$

where $q(z) = p_{\theta^{old}}(z|\mathbf{y},\mathbf{x})$. Subtracting $\log p_\theta(\mathbf{y}|\mathbf{x})$ and $\log p_{\theta^{old}}(\mathbf{y}|\mathbf{x})$, we have

$$\begin{aligned}&\log p_\theta(\mathbf{y}|\mathbf{x}) - \log p_{\theta^{old}}(\mathbf{y}|\mathbf{x})\\ &= \ell_{ELBO}(q,\theta) - \ell_{ELBO}(q,\theta^{old})\\ &\quad + KL[q(z)||p_\theta(z|\mathbf{y},\mathbf{x})] - KL[q(z)||p_{\theta^{old}}(z|\mathbf{y},\mathbf{x})].\end{aligned} \tag{11}$$

Given that $KL[q(z)||p_\theta(z|\mathbf{y},\mathbf{x})] \geq KL[q(z)||p_{\theta^{old}}(z|\mathbf{y},\mathbf{x})]$ and that $\ell_{ELBO}(q,\theta) - \ell_{ELBO}(q,\theta^{old}) = \mathbb{E}_{q(z)}[\log p_\theta(\mathbf{y},z|\mathbf{x})] - \mathbb{E}_{q(z)}[\log p_{\theta^{old}}(\mathbf{y},z|\mathbf{x})]$, we conclude that

$$\begin{aligned}&\log p_\theta(\mathbf{y}|\mathbf{x}) - \log p_{\theta^{old}}(\mathbf{y}|\mathbf{x})\\ &\geq \mathbb{E}_{q(z)}[\log p_\theta(\mathbf{y},z|\mathbf{x})] - \mathbb{E}_{q(z)}[\log p_{\theta^{old}}(\mathbf{y},z|\mathbf{x})] \geq 0\end{aligned} \tag{12}$$

because of the assumption $\mathbb{E}_{q(z)}[\log p_\theta(\mathbf{y},z|\mathbf{x})] \geq \mathbb{E}_{q(z)}[\log p_{\theta^{old}}(\mathbf{y},z|\mathbf{x})]$ [13]. □

Lemma 2. *Suppose that $\{\theta^{(e)}\}_{e=1}^{+\infty}$ denotes the sequence of trained model parameters from the maximisation of ℓ_{ELBO} in Eq. 6 such that:*

1. *the sequence $\{\log p_{\theta^{(e)}}(\mathbf{y}|\mathbf{x})\}_{e=1}^{+\infty}$ is bounded above, and*
2. *$\left(\mathbb{E}_{q(z)}[\log p_{\theta^{(e+1)}}(\mathbf{y},z|\mathbf{x})] - \mathbb{E}_{q(z)}[\log p_{\theta^{(e)}}(\mathbf{y},z|\mathbf{x})]\right) \geq \xi\left(\theta^{(e+1)} - \theta^{(e)}\right)^\top\left(\theta^{(e+1)} - \theta^{(e)}\right)$ for $\xi > 0$ and all $e \geq 1$, and $q(z) = p_{\theta^{(e)}}(z|\mathbf{y},\mathbf{x})$.*

Then the sequence $\{\theta^{(e)}\}_{e=1}^{+\infty}$ converges to some $\theta^\star \in \Theta$.

Proof. Following the proof for Theorem 2 in [13], the sequence $\{\log p_{\theta^{(e)}}(\mathbf{y}|\mathbf{x})\}_{e=1}^{+\infty}$ is non-decreasing (from Lemma 1) and bounded above (from assumption (1) in Lemma 2), so it converges to $L^\star < +\infty$. Therefore, according to Cauchy criterion [57], for any $\epsilon > 0$, we have $e^{(\epsilon)}$ such that, for $e \geq e^{(\epsilon)}$ and all $r \geq 1$,

$$\begin{aligned}&\sum_{j=1}^{r}\left(\log p_{\theta^{(e+j)}}(\mathbf{y}|\mathbf{x}) - \log p_{\theta^{(e+j-1)}}(\mathbf{y}|\mathbf{x})\right)\\ &= \left(\log p_{\theta^{(e+r)}}(\mathbf{y}|\mathbf{x}) - \log p_{\theta^{(e)}}(\mathbf{y}|\mathbf{x})\right) < \epsilon.\end{aligned} \tag{13}$$

From Eq. 12,

$$\begin{aligned}0 &\leq \mathbb{E}_{q(z)}[\log p_{\theta^{(e+j)}}(\mathbf{y},z|\mathbf{x})] - \mathbb{E}_{q(z)}[\log p_{\theta^{(e+j-1)}}(\mathbf{y},z|\mathbf{x})]\\ &\leq \log p_{\theta^{(e+j)}}(\mathbf{y}|\mathbf{x}) - \log p_{\theta^{(e+j-1)}}(\mathbf{y}|\mathbf{x})\end{aligned} \tag{14}$$

for $j \geq 1$ and $q(z) = p_{\theta^{(e+j-1)}}(z|\mathbf{y},\mathbf{x})$. Hence, from Eq. 13,

$$\sum_{j=1}^{r}\left(\mathbb{E}_{q(z)}[\log p_{\theta^{(e+j)}}(\mathbf{y},z|\mathbf{x})] - \mathbb{E}_{q(z)}[\log p_{\theta^{(e+j-1)}}(\mathbf{y},z|\mathbf{x})]\right) < \epsilon, \tag{15}$$

for $e \geq e^{(\epsilon)}$ and all $r \geq 1$. Given assumption (2) in Lemma 2 for $e, e+1, e+2, ..., e+r-1$, we have from Eq. 15,

$$\epsilon > \xi\sum_{j=1}^{r}\left(\theta^{(e+j)} - \theta^{(e+j-1)}\right)^\top\left(\theta^{(e+j)} - \theta^{(e+j-1)}\right), \tag{16}$$

so

$$\epsilon > \xi\left(\theta^{(e+r)} - \theta^{(e)}\right)^\top\left(\theta^{(e+r)} - \theta^{(e)}\right), \tag{17}$$

which is a requirement to prove the convergence of $\theta^{(e)}$ to some $\theta^\star \in \Theta$. □

4. Experiments

4.1. Experimental Setup

We evaluate our method on CIFAR-10/-100 [15], Controlled Noisy Web Labels (CNWL) [38], Clothing1M [17], and WebVision [18]. The CIFAR-10 and CIFAR-100 datasets contain 50,000 training images and 10,000 test images of size 32 × 32 pixels with 10 and 100 classes respectively. Since both these datasets have been annotated with clean labels, we use synthetic noise to evaluate the models. For CIFAR-10/-100, we evaluate three types of noise: symmetric [54,58], asymmetric [54,58], and semantic [50]. For symmetric noise we used $\eta \in \{0.2, 0.5, 0.8, 0.9\}$, where η was defined in Section Method as the symmetric noise probability. The asymmetric noise was applied to the dataset, similarly to [7], which replaces the labels *truck* → *automobile, bird* → *airplane, deer* → *horse*, and *cat* → *dog*. For asymmetric noise, we use the noise rates of 40% and 49%. For the semantic noise, we use the same setup from [50], which generates semantically noisy labels based on a trained VGG [59], DenseNet (DN) [60], and ResNet (RN) [61] on CIFAR-10 and CIFAR-100.

The CNWL dataset [38] is a benchmark to study real-world web label noise in a controlled setting. Both images and labels are crawled from the web and the noisy labels are determined by matching images. The controlled setting provide different magnitudes of label corruption in real applications, varying from 0 to 80%. CNWL provides controlled web noise for Mini-ImageNet dataset, called red noise. The red Mini-ImageNet consists of 50k training images and 5000 test images, with 100 classes. The original image sizes are of 84×84 pixels, which are resized to 32×32 pixels. The noise rates use in this work are 20%, 40%, 60% and 80%, as used in [16].

Clothing1M is a dataset of 14 classes containing 1 million training images downloaded from online shopping websites. All training images are resized to 256 × 256 pixels [7,62]. The noise rate is estimated to be asymmetric [53] with a rate of 40% [17] and class distribution is heavily imbalanced. Clothing1M has 50k and 14k clean images for training and validation, respectively, but we do not use them for training. The testing set has 10k clean-labelled images.

The WebVision [18] is a real-world large scale dataset containing 2.4 million images collected from the internet, with the same 1000 classes from ImageNet [63]. As the images vary in size, we resized them to 227 × 227 pixels. WebVision provides a clean test set of 50k images, with 50 images per class. We compare our model using the first 50 classes of the Google image subset, as in [7,64]

All experiments were run on Intel Core i9 computer with 128GB memory and 4x nVidia GeForce RTX 3090.

4.2. Implementation

CIFAR-10/-100 We use PreAct-ResNet-18 as our backbone model following [7]. For the self-supervised pre-training learning task, we adopt the standard SimCLR [11] implementation with a batch size of 512, SGD optimiser with a learning rate of 0.4, decay rate of 0.1,

Table 1
Test accuracy (%) for all competing methods on CIFAR-10 and CIFAR-100 under symmetric and asymmetric noises. Results from related approaches are as presented in [7]. The results with (*) were produced by locally running the published code provided by the authors. Top methods within 1% are in **bold**.

dataset		CIFAR-10						CIFAR-100			
Noise type		sym.				asym.		sym.			
Method/ noise ratio		20%	50%	80%	90%	40%	49%	20%	50%	80%	90%
Cross-Entropy	Best	86.8	79.4	62.9	42.7	85.0	-	62.0	46.7	19.9	10.1
	Last	82.7	57.9	26.1	16.8	72.3	-	61.8	37.3	8.8	3.5
Coteaching+ [33]	Best	89.5	85.7	67.4	47.9	-	-	65.6	51.8	27.9	13.7
	Last	88.2	84.1	45.5	30.1	-	-	64.1	45.3	15.5	8.8
MixUp [43]	Best	**95.6**	87.1	71.6	52.2	-	-	67.8	57.3	30.8	14.6
	Last	92.3	77.3	46.7	43.9	-	-	66.0	46.6	17.6	8.1
PENCIL [53]	Best	92.4	89.1	77.5	58.9	88.5	-	69.4	57.5	31.1	15.3
	Last	92.0	88.7	76.1	58.2	88.1	-	68.1	56.4	20.7	8.8
Meta-Learning [14]	Best	92.9	89.3	77.4	58.7	89.2	-	68.5	59.2	42.4	19.5
	Last	92.0	88.8	76.1	58.3	88.6	-	67.7	58.0	40.1	14.3
M-correction [3]	Best	94.0	92.0	86.8	69.1	87.4	-	73.9	66.1	48.2	24.3
	Last	93.8	91.9	86.6	68.7	86.3	-	73.4	65.4	47.6	20.5
MentorMix [38]	Best	**95.6**	-	81.0	-	-	-	**78.6**	-	41.2	-
	Last	-	-	-	-	-	-	-	-	-	-
MOIT+ [6]	Best	94.1	-	75.8	-	**93.3**	-	75.9	-	51.4	-
	Last	-	-	-	-	-	-	-	-	-	-
DivideMix [7]	Best	**96.1**	**94.6**	**93.2**	76.0	**93.4**	83.7*	77.3	74.6	60.2	31.5
	Last	**95.7**	**94.4**	**92.9**	75.4	92.1	76.3*	**76.9**	74.2	59.6	31.0
ELR+ [22]	Best	**95.8**	**94.8**	**93.3**	78.7	**93.0**	-	**77.6**	73.6	60.8	33.4
	Last	-	-	-	-	-	-	-	-	-	-
PES [55]	Best	**95.9**	**95.1**	**93.1**	-	77.4	-	74.3	61.6	-	-
	Last	-	-	-	-	-	-	-	-	-	-
FSR [56]	Best	**95.1**	-	82.8	-	**93.6**	-	**78.7**	-	46.7	-
	Last	-	-	-	-	-	-	-	-	-	-
DRPL [5]	Best	**94.2**	-	64.4	-	**93.1**	-	71.3	-	53.0	-
	Last	-	-	-	-	-	-	-	-	-	-
ScanMix (Ours)	Best	**96.0**	**94.5**	**93.5**	**91.0**	**93.7**	**88.7**	77.0	**75.7**	**66.0**	**58.5**
	Last	**95.7**	**93.9**	**92.6**	**90.3**	**93.4**	**87.1**	**76.0**	**75.4**	**65.0**	**58.2**

momentum of 0.9 and weight decay of 0.0001, and run it for 500 epochs. This pre-trained model produces feature representations of 128 dimensions. Using these representations we mine $K = 20$ nearest neighbours (as in [10]) for each sample to form the sets $\{\mathcal{N}_{\mathbf{x}_i}\}_{i=1}^{|\mathcal{D}|}$, defined in the Method Section. For the semantic clustering task, we use a batch size of 128, $\lambda_e = 2$ as in [10], SGD optimiser with momentum of 0.9, weight decay of 0.0005 and learning rate $\in \{0.001, 0.00001\}$ based on the predicted noise rate, which is estimated with $|\mathcal{U}|/|\mathcal{D}|$, defined in Eq. 8 – if this ratio is larger than 0.6, then the learning rate is 0.001, otherwise, the learning rate is 0.00001. This accounts for the fact that when the estimated label noise is high, then we want to increase the influence of semantic clustering in the training; but when the label noise is low, then the signal from the labels in the SSL method should carry more weighting. For the SSL, we adopt the implementation of [7] and use the same hyperparameters, where we rely on SGD with learning rate of 0.02 (which is reduced to 0.002 halfway through the training), momentum of 0.9 and weight decay of 0.0005. Number of epochs $E = 300$.

Red Mini-ImageNet We use PreAct-ResNet-18 as our backbone model, following [16]. For the self-supervised pre-training, we adopt the standard SimCLR [11] implementation with batch size 128. All other parameters for the self-supervised pre-training and semantic clustering are the same as for CIFAR, except for the semantic clustering learning rate, which we used 0.001, and the $\lambda_u = 0$ for all noise rates. The feature representation learned from this process has 128 dimensions. For the SSL, we adopt the implementation of [16], where we train for 300 epochs, relying on SGD with learning rate of 0.02 (decreased by a factor of ten at epoch 200 and epoch 250), momentum of 0.9 and weight decay of 5e-4. We also resized the images from 84×84 to 32×32 [16].

Clothing1M We use ResNet-50 as our backbone model, which is trained for 80 epochs with a WarmUp stage of 1 epoch. For the self-supervised pre-training task we adopt the standard MoCo-v2 method for a 4-GPU training [9] with a batch size of 128, SGD optimiser with a learning rate of 0.015, momentum of 0.9 and weight decay of 0.0001 and run it for 100 epochs. In this pre-training task we use 100k randomly selected images from Clothing1M training set as the pre-training images. All the other parameters were the same as described above for CIFAR, except for the batch size of semantic clustering task was 64 and the number of epochs E=80. During ScanMix training, we followed [7], which relies on 64k randomly selected training images from the entire training for each epoch. As the training images change for every epoch, we adapted ScanMix to update the nearest neighbors before training each batch. Different from [7], we do not use the pre-trained weights from ImageNet.

WebVision We use InceptionResNet-V2 as our backbone model, following [7]. For the self-supervised pre-training task we adopt the standard MoCo-v2 method for a 4-GPU training [9] with a batch size of 128, SGD optimiser with a learning rate of 0.015, momentum of 0.9 and weight decay of 0.0001, and run it for 100 epochs with a WarmUp stage of 1 epoch. The feature representations learned from this process have 128 dimensions. All the other parameters were the same as described above for CIFAR, except the batch size of semantic clustering task was 64, and number of epochs $E = 100$.

4.3. Comparison with State-of-the-Art

We compare ScanMix with several existing methods using the datasets described in Sec. Experimental Setup. For CIFAR-10 and CIFAR-100 in Table 1, we evaluate the models using different levels of symmetric label noise, ranging from 20% to 90% and asymmetric noise rates of 40% and 49%. We report both the best test accuracy across all epochs and the averaged test accuracy over the

Table 2
Test accuracy (%) for Semantic Noise. Results from baseline methods are as presented in [50]. The results with (*) were produced by locally running the published code provided by the authors. Top methods within 1% are in **bold**.

dataset	CIFAR-10			CIFAR-100		
Method/ noise ratio	DenseNet (32%)	ResNet (38%)	VGG (34%)	DenseNet (34%)	ResNet (37%)	VGG (37%)
D2L + RoG [50]	68.57	60.25	59.94	31.67	39.92	45.42
CE + RoG [50]	68.33	64.15	70.04	61.14	53.09	53.64
Bootstrap + RoG [50]	68.38	64.03	70.11	54.71	53.30	53.76
Forward + RoG [50]	68.20	64.24	70.09	53.91	53.36	53.63
Backward + RoG [50]	68.66	63.45	70.18	54.01	53.03	53.50
DivideMix* [7]	84.57	81.61	85.71	**68.40**	66.28	**66.84**
ScanMix (Ours)	**89.70**	**85.58**	**89.96**	**68.44**	**67.36**	**67.34**

Table 3
Test accuracy (%) for Red Mini-ImageNet. Results from baseline methods are as presented in [16]. Top methods within 1% are in **bold**.

Method/ noise ratio	20%	40%	60%	80%
Cross-entropy [16]	47.36	42.70	37.30	29.76
Mixup [43]	49.10	46.40	40.58	33.58
DivideMix [7]	50.96	46.72	43.14	34.50
MentorMix [38]	51.02	47.14	43.80	33.46
FaMUS [16]	51.42	48.06	45.10	35.50
ScanMix (Ours)	**59.06**	**54.54**	**52.36**	**40.00**

last 10 epochs of training. Results show that our method significantly outperforms the previous methods under severe label noise. Specifically, we observe an increase of roughly +13% for CIFAR-10 with 90% symmetric noise, +5% for CIFAR-10 with 49% asymmetric noise, +25% for CIFAR-100 with 90% symmetric noise and +5% for CIFAR-100 with 80% symmetric noise. These results show that our ScanMix does make the model more robust to noisy labels than previous methods, particularly for severe label noise. To demonstrate that more clearly, we computed mean and variance accuracy using bootstrapping, and applied a T-test to compare ScanMix and DivideMix. For all cases that we claim to be better in Table 1 (symmetric at 80% and 90% on Cifar10,100 and asymmetric at 40% and 49% on Cifar10), we obtained p-values < 0.01.

Table 2 shows the ability of our method to handle semantic noise, which can be regarded as a harder and more realistic type of label noise that depends not only on label transition, but also on the image features. The current SOTA for this benchmark is RoG [50], and even though the noise rates are not particularly large, our ScanMix shows results that are better by a large margin varying from 12% to 22%. The results on Red Mini-ImageNet [16] in Table 3 shows that ScanMix provides substantial gains form 4% to 7% over the SOTA for all noise rates.

We also evaluate ScanMix on the noisy large-scale dataset WebVision. Table 4 shows the Top-1/-5 test accuracy using the WebVision and ILSVRC12 test sets. Results show that ScanMix is slightly better than the SOTA for both WebVision test sets and top-5 ILSVRC12 test set. This suggests that our approach is also effective in large-scale, low noise rate problems. Results on Clothing1M in Tab. 5 show that ScanMix is on par with the current SOTA in the field, even though our method does use the whole training set for the pre-training stage (recall that we randomly selected 100k out of the 1M training images for pre-training) and differently from most of previous approaches, we do not rely on an ImageNet pre-trained model, as explained in Sec. 4.2. These two issues should have had a significant negative impact on the performance of ScanMix, but these Clothing1M results indicate that ScanMix remained robust in this challenging scenario.

For the running time complexity, ScanMix and DivideMix are similar asymptotically since both have linear complexity in terms of the training set size, as described in the Section 3.2. In practice, ScanMix is two times slower. On CIFAR-10, DivideMix takes 13.93 GPU hours while ScanMix takes 27.94 hours (where pre-train takes 5.9 GPU hours).

Table 4
Test accuracy (%) for WebVision [18] by methods trained with 100 epochs. Baseline results are as presented in [7]. Top methods within 1% are in **bold**.

dataset	WebVision		ILSVRC12	
Method	Top-1	Top-5	Top-1	Top-5
F-correction [49]	61.12	82.68	57.36	82.36
Decoupling [31]	62.54	84.74	58.26	82.26
D2L [34]	62.68	84.00	57.80	81.36
MentorNet [30]	63.00	81.40	57.80	79.92
Co-teaching [32]	63.58	85.20	61.48	84.70
Iterative-CV [64]	65.24	85.34	61.60	84.98
MentorMix [38]	76.00	90.20	72.90	91.10
DivideMix [7]	77.32	91.64	**75.20**	90.84
ELR+ [22]	77.78	91.68	70.29	89.76
MOIT+ [6]	78.76	-	-	-
FSR [56]	74.90	88.20	72.30	87.20
ScanMix (Ours)	**80.04**	**93.04**	**75.76**	**92.60**

Table 5
Results on Clothing1M [17] for ScanMix and SOTA approaches (SOTA results collected from [7] or original papers). Top results within 1% are highlighted in **bold**.

Method	Test Accuracy
Cross-Entropy [7]	69.21
M-correction [51]	71.00
Meta-Cleaner [37]	72.50
Meta-Learning [54]	73.47
PENCIL[53]	73.49
DeepSelf [62]	**74.45**
CleanNet [65]	**74.69**
DivideMix [7]	**74.76**
ScanMix (Ours)	**74.35**

4.4. Ablation Study

We show the results of the ablation study of ScanMix in Table 6. Using classification accuracy in the testing set of CIFAR-10 and CIFAR-100 under symmetric and asymmetric noises at several rates, we aim to show the influence of self-supervised training by itself or in combination with SSL. For self-supervised learning, we use the current SOTA method, SCAN [10], displayed in the first two rows, with the first row containing the published results, and the second, our replicated results using the authors' code. The result is the same across different noise rates because it never uses the noisy labels for training. Using the pure SSL method, DivideMix [7], which is the current SOTA in noisy label learning, we see that it has much better results for low noise levels, but SCAN is better for severe label noise. When using SCAN for pre-training DivideMix,

Table 6
In this ablation study we show the classification accuracy in the testing set of CIFAR-10 and CIFAR-100 under symmetric and asymmetric noises at several rates. First, we show the results of self-supervised pre-training using the current SOTA SCAN [10] (first two rows, with the results with (*) produced by locally running the published code provided by the authors). Then we show the current SOTA SSL learning for noisy label DivideMix [7]. Next, we show the results of DivideMix pre-trained with SCAN. The last row shows our ScanMix that combines SSL and semantic clustering. The top results within 1% are highlighted in **bold**.

dataset	CIFAR-10						CIFAR-100			
Noise type	sym.				asym.		sym.			
Method/ noise ratio	20%	50%	80%	90%	40%	49%	20%	50%	80%	90%
Self-superv. pre-train =(SCAN) [10]	81.6	81.6	81.6	81.6	81.6	81.6	44.0	44.0	44.0	44.0
Self-superv. pre-train* (SCAN) [10]	77.5	77.5	77.5	77.5	77.5	77.5	37.1	37.1	37.1	37.1
SSL (DivideMix) [7]	**96.1**	**94.6**	**93.2**	76.0	**93.4**	83.7*	**77.3**	74.6	60.2	31.5
Self-superv. pre-train + SSL (DivideMix)*	**95.3**	**94.4**	**93.7**	**91.0**	**93.3**	85.9	75.2	74.4	64.4	52.8
ScanMix (Ours)	**96.0**	**94.5**	**93.5**	**91.0**	**93.7**	**88.6**	**77.0**	**75.7**	**66.0**	**58.5**

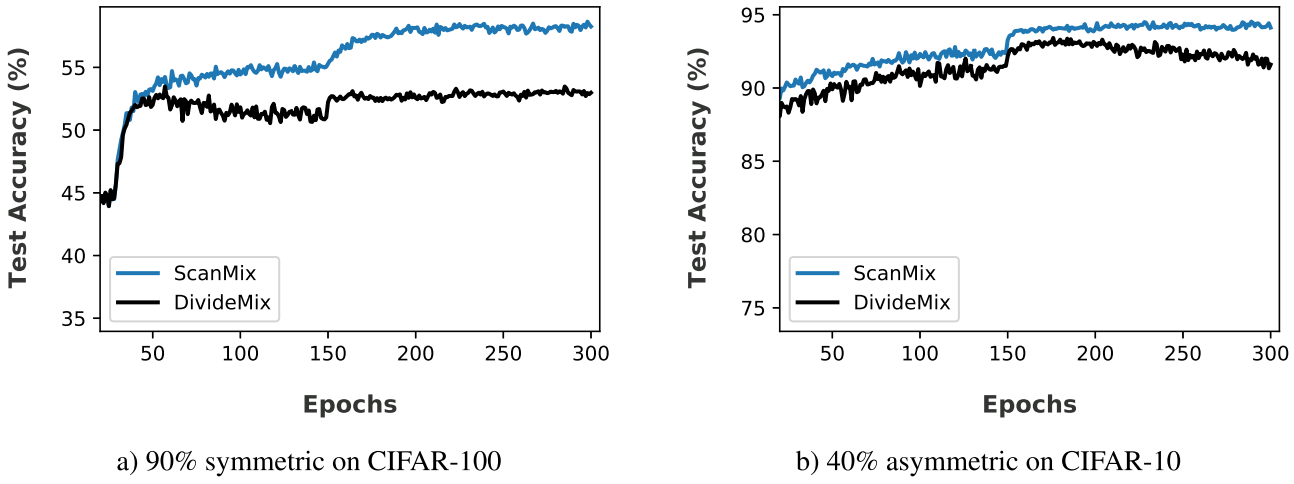

Fig. 3. Test accuracy (%) as a function of the number of training epochs for ScanMix (blue) and DivideMix (black) for 90% asymmetric noise on CIFAR-100 (a), and 40% asymmetric noise on CIFAR-10 (b).

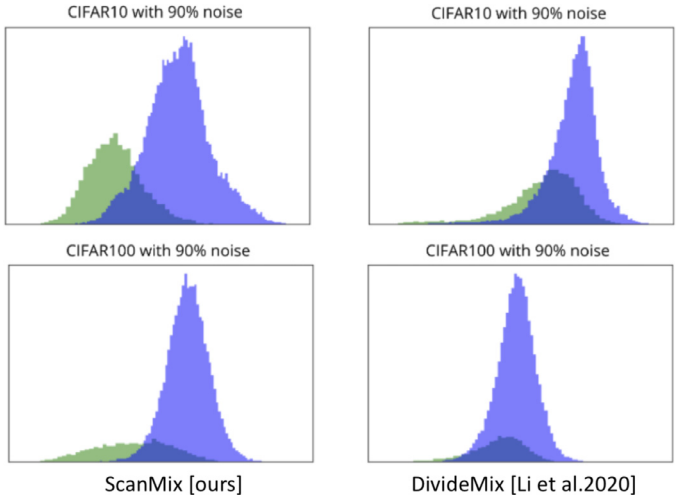

Fig. 4. Per-sample normalised loss distributions of the training set produced in the early stages of the training by our ScanMix (left) and DivideMix [7] (right) for CIFAR-10 (top) and CIFAR-100 (bottom) affected by 90% label noise, where green bars represent the clean samples and blue bars the noisy samples.

we note that results become quite good for all types and levels of noise. Neverthless, our ScanMix improves the results of SCAN + DivideMix, showing the efficacy of ScanMix, which combines SSL with semantic clustering.

A common issue with learning with noisy labels is the tendency of models to overfit the noisy labels during training [22,34], causing a reduction of accuracy in the test set. To show the robustness of ScanMix to this issue, we present in Fig. 3 the prediction accuracy on the test set as a function of the number of training epochs for ScanMix (blue) and DivideMix (black) for 90% asymmetric noise on CIFAR-100 (a), and 40% asymmetric noise on CIFAR-10 (b). Notice that in both cases, ScanMix is shown to be more robust to overfitting than DivideMix.

We also demonstrate that ScanMix is able to provide a reliable separation between clean and noisy samples. Figure 4 shows a comparison between the distributions of losses produced by ScanMix and DivideMix at one of the early epochs of training for CIFAR-10 and CIFAR-100 affected by 90% label noise. This figure shows that semantic clustering combined with SSL in ScanMix enables a much clearer separation between the clean (green bars) and noisy (blue bars) samples, when compared with the distribution produced by DivideMix. Such clearer separation will help the classification of clean samples in Eq. 8, which in turn will improve the performance of the SSL in Eq. 7.

5. Conclusion and Future Work

In this work we presented ScanMix, a novel training strategy that produces superior robustness to severe label noise and competitive robustness to non-severe label noise problems, compared with the SOTA. Results on CIFAR-10/-100, Red Mini-ImageNet, Clothing1M and WebVision showed that our proposed ScanMix outperformed SOTA methods, with large improvements particularly in severe label noise problems. Our approach also produced superior results for semantic noise and real-world web label noise, which are regarded to be the most challenging noise types. These results show evidence for our claims in Section 1, that SSL noisy

label learning methods (e.g., DivideMix [7]) depend on an effective way to classify clean and noisy samples, which works well for small noise rates, but not for severe noise rates. Semantic clustering methods (e.g., SCAN [10]) ignore labels, enabling them to work well for severe noise rates, but poorly for low noise. Hence, our ScanMix explores the advantages of SSL and semantic clustering to achieve SOTA results for severe label noise rates, while being competitive for non-severe label noise.

The increasing availability of large-scale datasets is associated with a decreasing availability of trustworthy annotations. This can introduce label noise into training sets, and reduce the generalisation ability of machine learning models. Our method can mitigate this issue and enable the use of large-scale datasets by communities that do not have other ways to re-annotate such datasets, thus democratising machine learning. A drawback of our approach is the longer training time, compared with the SOTA DivideMix [7], so we are currently working on an approach that mitigates this issue by having a joint self-supervised and semi-supervised training algorithm. Another point that can be improved in ScanMix is the semantic clustering algorithm, which can explore more robust methods, such as RBSMF [45], MPF [44], ClusterNet [46], and US-ADTM [47].

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by Australian Research Council through grant FT190100525.

References

[1] G. Litjens, T. Kooi, B.E. Bejnordi, A.A.A. Setio, F. Ciompi, M. Ghafoorian, J.A. Van Der Laak, B. Van Ginneken, C.I. Sánchez, A survey on deep learning in medical image analysis, Medical image analysis 42 (2017) 60–88.
[2] B. Frénay, M. Verleysen, Classification in the presence of label noise: a survey, IEEE transactions on neural networks and learning systems 25 (5) (2013) 845–869.
[3] C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Vinyals, Understanding deep learning requires rethinking generalization, in: International Conference on Learning Representations (ICLR), 2017.
[4] Y. Ding, L. Wang, D. Fan, B. Gong, A semi-supervised two-stage approach to learning from noisy labels, in: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2018, pp. 1215–1224.
[5] D. Ortego, E. Arazo, P. Albert, N.E. O'Connor, K. McGuinness, Towards robust learning with different label noise distributions, in: 2020 25th International Conference on Pattern Recognition (ICPR), IEEE, 2021, pp. 7020–7027.
[6] D. Ortego, E. Arazo, P. Albert, N.E. O'Connor, K. McGuinness, Multi-objective interpolation training for robustness to label noise, in: Conference on Computer Vision and Pattern Recognition (CVPR), 2021.
[7] J. Li, R. Socher, S.C. Hoi, Dividemix: Learning with noisy labels as semi-supervised learning, in: International Conference on Learning Representations (ICLR), 2020.
[8] K. He, H. Fan, Y. Wu, S. Xie, R. Girshick, Momentum contrast for unsupervised visual representation learning, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9729–9738.
[9] X. Chen, H. Fan, R. Girshick, K. He, Improved Baselines with Momentum Contrastive Learning. 2003.04297.
[10] W. Van Gansbeke, S. Vandenhende, S. Georgoulis, M. Proesmans, L. Van Gool, Scan: Learning to classify images without labels, in: European Conference on Computer Vision, Springer, 2020, pp. 268–285.
[11] T. Chen, S. Kornblith, M. Norouzi, G. Hinton, A simple framework for contrastive learning of visual representations, in: International Conference on Machine Learning (ICML), PMLR, 2020, pp. 1597–1607.
[12] T. Nguyen, C. Mummadi, T. Ngo, L. Beggel, T. Brox, Self: learning to filter noisy labels with self-ensembling, in: International Conference on Learning Representations (ICLR), 2020.
[13] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the em algorithm, Journal of the Royal Statistical Society: Series B (Methodological) 39 (1) (1977) 1–22.
[14] U. Rebbapragada, C.E. Brodley, Class noise mitigation through instance weighting, in: European conference on machine learning, Springer, 2007, pp. 708–715.
[15] A. Krizhevsky, G. Hinton, et al., Learning multiple layers of features from tiny images, Citeseer, 2009.
[16] Y. Xu, L. Zhu, L. Jiang, Y. Yang, Faster meta update strategy for noise-robust deep learning, in: Conference on Computer Vision and Pattern Recognition (CVPR), 2021.
[17] T. Xiao, T. Xia, Y. Yang, C. Huang, 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.
[18] W. Li, L. Wang, W. Li, E. Agustsson, L. Van Gool, Webvision database: Visual learning and understanding from web data, arXiv preprint arXiv:1708.02862, 2017.
[19] L. Jaehwan, Y. Donggeun, 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.
[20] B. Yuan, J. Chen, W. Zhang, H.-S. Tai, S. McMains, Iterative cross learning on noisy labels, in: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), IEEE, 2018, pp. 757–765.
[21] Y. Zhang, S. Zheng, P. Wu, M. Goswami, C. Chen, Learning with feature dependent label noise: a progressive approach, in: International Conference on Feature Representation (ICLR), 2021.
[22] S. Liu, J. Niles-Weed, N. Razavian, C. Fernandez-Granda, Early-learning regularization prevents memorization of noisy labels, in: Conference on Neural Information Processing Systems (NeurIPS), 2020.
[23] X. Wang, Y. Hua, E. Kodirov, 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.
[24] Y. Wang, X. Ma, Z. Chen, Y. Luo, J. Yi, 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.
[25] B. Han, G. Niu, J. Yao, X. Yu, M. Xu, I. Tsang, M. Sugiyama, Pumpout: A meta approach for robustly training deep neural networks with noisy labels, ICML Workshop on Uncertainty and Robustness in Deep Learning, 2019.
[26] H. Sun, C. Guo, Q. Wei, Z. Han, Y. Yin, Learning to rectify for robust learning with noisy labels, Pattern Recognition 124 (2022) 108467.
[27] M. Ren, W. Zeng, B. Yang, R. Urtasun, Learning to reweight examples for robust deep learning, in: International conference on machine learning, PMLR, 2018, pp. 4334–4343.
[28] Q. Miao, Y. Cao, G. Xia, M. Gong, J. Liu, 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 27 (11) (2015) 2216–2228.
[29] A. Tarvainen, H. Valpola, Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results, in: Advances in neural information processing systems, 2017, pp. 1195–1204.
[30] L. Jiang, Z. Zhou, T. Leung, L.-J. Li, 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.
[31] E. Malach, S. Shalev-Shwartz, Decoupling" when to update" from" how to update", in: Advances in Neural Information Processing Systems, 2017, pp. 960–970.
[32] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, 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.
[33] X. Yu, B. Han, J. Yao, G. Niu, I. Tsang, M. Sugiyama, How does disagreement help generalization against label corruption? in: International Conference on Machine Learning, PMLR, 2019, pp. 7164–7173.
[34] X. Ma, Y. Wang, M.E. Houle, S. Zhou, S. Erfani, S. Xia, S. Wijewickrema, J. Bailey, Dimensionality-driven learning with noisy labels, in: International Conference on Machine Learning, 2018, pp. 3355–3364.
[35] X. Yu, T. Liu, M. Gong, D. Tao, Learning with biased complementary labels, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 68–83.
[36] Y. Kim, J. Yim, J. Yun, J. Kim, Nlnl: Negative learning for noisy labels, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 101–110.
[37] W. Zhang, Y. Wang, 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.
[38] L. Jiang, D. Huang, M. Liu, W. Yang, Beyond synthetic noise: Deep learning on controlled noisy labels, International Conference on Machine Learning (ICML), 2020.
[39] Z. Zhang, H. Zhang, S.O. Arik, H. Lee, 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.
[40] R. Sachdeva, F.R. Cordeiro, V. Belagiannis, I. Reid, G. Carneiro, Evidentialmix: Learning with combined open-set and closed-set noisy labels, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 3607–3615.
[41] F.R. Cordeiro, R. Sachdeva, V. Belagiannis, I. Reid, G. Carneiro, Longremix: Robust learning with high confidence samples in a noisy label environment, arXiv preprint arXiv:2103.04173 (2021).
[42] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, C. Raffel, Mixmatch: A holistic approach to semi-supervised learning, Neural Information Processing Systems (NeurIPS), 2019.
[43] H. Zhang, M. Cisse, Y.N. Dauphin, D. Lopez-Paz, mixup: Beyond empirical risk

minimization, in: International Conference on Learning Representations (ICLR), 2018.
[44] Q. Wang, M. Chen, F. Nie, X. Li, Detecting coherent groups in crowd scenes by multiview clustering, IEEE transactions on pattern analysis and machine intelligence 42 (1) (2018) 46–58.
[45] Q. Wang, X. He, X. Jiang, X. Li, Robust bi-stochastic graph regularized matrix factorization for data clustering, IEEE Transactions on Pattern Analysis and Machine Intelligence 44 (1) (2020) 390–403.
[46] A. Shukla, G.S. Cheema, S. Anand, Semi-supervised clustering with neural networks, in: 2020 IEEE Sixth International Conference on Multimedia Big Data (BigMM), IEEE, 2020, pp. 152–161.
[47] T. Han, J. Gao, Y. Yuan, Q. Wang, Unsupervised semantic aggregation and deformable template matching for semi-supervised learning, Advances in Neural Information Processing Systems 33 (2020) 9972–9982.
[48] F. Chiaroni, M.-C. Rahal, N. Hueber, F. Dufaux, Hallucinating a cleanly labeled augmented dataset from a noisy labeled dataset using gan, in: 2019 IEEE International Conference on Image Processing (ICIP), IEEE, 2019, pp. 3616–3620.
[49] G. Patrini, A. Rozza, A. Krishna Menon, R. Nock, 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.
[50] K. Lee, S. Yun, K. Lee, H. Lee, B. Li, J. Shin, Robust inference via generative classifiers for handling noisy labels, in: International Conference on Machine Learning, PMLR, 2019, pp. 3763–3772.
[51] E. Arazo, D. Ortego, P. Albert, N. O'Connor, K. Mcguinness, Unsupervised label noise modeling and loss correction, in: International Conference on Machine Learning, 2019, pp. 312–321.
[52] S. Zhang, M. Bansal, Addressing semantic drift in question generation for semi–supervised question answering, arXiv preprint arXiv:1909.06356, 2019.
[53] K. Yi, 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.
[54] J. Li, Y. Wong, Q. Zhao, 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.
[55] Y. Bai, E. Yang, B. Han, Y. Yang, J. Li, Y. Mao, G. Niu, T. Liu, Understanding and improving early stopping for learning with noisy labels, Advances in Neural Information Processing Systems, volume 34, 2021.
[56] Z. Zhang, T. Pfister, Learning fast sample re-weighting without reward data, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 725–734.
[57] L. Nguyen, Tutorial on EM algorithm, Technical Report, 2020.
[58] D. Tanaka, D. Ikami, T. Yamasaki, K. Aizawa, Joint optimization framework for learning with noisy labels, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 5552–5560.
[59] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, International Conference on Learning Representations (ICLR), 2015.
[60] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: IEEE conference on computer vision and pattern recognition, 2017, pp. 4700–4708.
[61] K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks, in: European conference on computer vision, Springer, 2016, pp. 630–645.
[62] J. Han, P. Luo, X. Wang, Deep self-learning from noisy labels, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 5138–5147.
[63] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, 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.
[64] P. Chen, B.B. Liao, G. Chen, S. Zhang, Understanding and utilizing deep neural networks trained with noisy labels, in: International Conference on Machine Learning, PMLR, 2019, pp. 1062–1070.
[65] K.-H. Lee, X. He, L. Zhang, 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.

Ragav Sachdeva is a PhD student in the Visual Geometry Group at the University of Oxford, supervised by Prof. Andrew Zisserman. He obtained his undergraduate degree in computer science at the University of Adelaide, where he did his honours thesis with Prof. Gustavo Carneiro.

Filipe R. Cordeiro is a professor of the Department of Computing at Universidade Federal Rural de Pernambuco (UFRPE). In 2015, he received his Ph.D. in computer science from the Federal University of Pernambuco (UFPE). Filipe's mains contributions are in the area of computer vision, medical image analysis, and machine learning.

Vasileios Belagiannis is a professor in the Faculty of Computer Science at Otto von Guericke University Magdeburg. His research deals with topics such as representation learning, uncertainty estimation, multi-modal learning, learning with different forms of supervision, learning algorithm for noisy labels, few-shot learning and meta-learning.

Ian Reid is the Head of the School of Computer Science at the University of Adelaide, and the senior researcher at the Australian Institute for Machine Learning. His research interests include robotic and active vision, visual tracking, SLAM, human motion capture and intelligent visual surveillance.

Gustavo Carneiro is a professor in the School of Computer Science at the University of Adelaide, Director of Medical Machine Learning at the Australian Institute of Machine Learning and an Australian Research Council Future Fellow. His main research interests are in computer vision, medical image analysis and machine learning. He is moving to the CVSSP at the University of Surrey in December 2022.