SSR: An Efficient and Robust Framework for Learning with Unknown Label Noise

Chen Feng
https://sites.google.com/view/mr-chenfeng
Georgios Tzimiropoulos
https://ytzimiro.github.io/
Ioannis Patras
https://sites.google.com/view/ioannispatras

School of Electronic Engineering and Computer Science
Queen Mary University of London
London, UK

Abstract

Despite the large progress in supervised learning with neural networks, there are significant challenges in obtaining high-quality, large-scale and accurately labelled datasets. In such a context, how to learn in the presence of noisy labels has received more and more attention. As a relatively complex problem, in order to achieve good results, current approaches often integrate components from several fields, such as supervised learning, semi-supervised learning, transfer learning and resulting in complicated methods. Furthermore, they often make multiple assumptions about the type of noise of the data. This affects the model robustness and limits its performance under different noise conditions. In this paper, we consider a novel problem setting, ***L**earning with **U**nknown **L**abel **N**oise* (**LULN**), that is, learning when both the degree and the type of noise are unknown. Under this setting, unlike previous methods that often introduce multiple assumptions and lead to complex solutions, we propose a simple, efficient and robust framework named ***S**ample **S**election and **R**elabelling* (**SSR**), that with a minimal number of hyperparameters achieves SOTA results in various conditions. At the heart of our method is a sample selection and relabelling mechanism based on a **n**on-**p**arametric **K**NN classifier (NPK) g_q and a **p**arametric **m**odel **c**lassifier (PMC) g_p, respectively, to select the clean samples and gradually relabel the noisy samples. Without bells and whistles, such as model co-training, self-supervised pre-training and semi-supervised learning, and with robustness concerning the settings of its few hyper-parameters, our method significantly surpasses previous methods on both CIFAR10/CIFAR100 with synthetic noise and real-world noisy datasets such as WebVision, Clothing1M and ANIMAL-10N. Code is available at https://github.com/MrChenFeng/SSR_BMVC2022.

1 Introduction

It is now commonly accepted that supervised learning with deep neural networks can provide excellent solutions for a wide range of problems, so long as there is sufficient availability of labelled training data and computational resources. However, these results have been mostly obtained using well-curated datasets in which the labels are of high quality. In the real world, it is often costly to obtain high-quality labels, especially for large-scale datasets. A common

approach is to use semi-automatic methods to obtain the labels (e.g. "webly-labelled" images where the images and labels are obtained by web-crawling). While such methods can greatly reduce the time and cost of manual labelling, they also lead to low-quality noisy labels.

In such settings, noise is one of the following two types: closed-set noise where the true labels belong to one of the given classes (Set B in fig. 1) and open-set noise where the true labels do not belong to the set of labels of the classification problem (Set C in fig. 1). To deal with different types of noise, two main types of methods have been proposed, which we name here as probability-consistent methods and probability-approximate methods.

Probability-consistent methods usually model noise patterns directly and propose corresponding probabilistic adjustment techniques, e.g., robust loss functions [8, 28, 33] and noise corrections based on noise transition matrix [9]. However, accurate modelling of noise patterns is non-trivial, and often cannot even model open-set noise. Also, due to the necessary simplifications of probabilistic modelling, such methods often perform poorly with heavy and complex noise. More recently, probability-approximate methods, that is methods that do not model the noise patterns explicitly become perhaps the dominant paradigm, especially ones that are based on sample selection. Earlier methods often reduce the influence of noise samples by selecting a clean subset and training only with it [10, 12, 18, 36]. Recent methods tend to further employ semi-supervised learning methods, such as MixMatch [4], to fully explore the entire dataset by treating the selected clean subset as labelled samples and the non-selected subset as unlabeled samples [14, 20]. These methods, generally, do not consider the presence of open-set noise in the dataset, since most current semi-supervised learning methods can not deal with open-set noise appropriately. To address this, several methods [23, 30] extend the sample selection idea by further identifying the open-set noise and excluding it from the semi-supervised training.

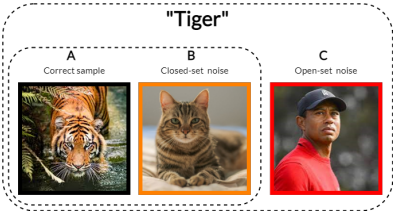

Figure 1: Different "tigers".

In general, the above methods make assumptions about the pattern of the noise, such as the confidence penalty specifically for asymmetric noise in DivideMix [14]. However, these mechanisms are often detrimental when the noise pattern does not meet the assumptions – for example, explicitly filtering open-set noise in the absence of open-set noise may result in clean hard samples being removed. Furthermore, due to the complexity of combining multiple modules, the above methods usually need to adjust complex hyperparameters according to the type and degree of noise.

In this paper, we consider a novel problem setting — ***L**earning with **U**nknown **L**abel **N**oise* (**LULN**), that is, learning when both the degree and the type of noise are unknown. Striving for simplicity and robustness, we propose a simple method for **LUNL**, namely ***S**ample **S**election and **R**elabelling* (**SSR**) (section 3.2), with two components that are clearly decoupled: a selection mechanism that identifies clean samples with correct labels, and a relabelling mechanism that aims to recover correct labels of wrongly labelled noisy samples. These two major components are based on the two simple and necessary assumptions for **LULN**, namely, that samples with highly-consistent annotations with their neighbours are often clean, and that very confident model predictions are often trustworthy. Once a well-labelled subset is constructed this way we use the most basic supervised training scheme with a cross-entropy loss. Optionally, a feature consistency loss can be used for all data so

as to deal better with open-set noise.

Without bells and whistles, such as semi-supervised learning, self-supervised model pre-training and model co-training, our method is shown to be robust to the values of its very few hyperparameters through extensive experiments and ablation studies and to consistently outperform the state-of-the-art by a large margin in various datasets.

2 Related Works

This paper mainly focuses on the probability-approximate methods, especially methods based on sample selection. For a detailed introduction to probability-consistent methods described above, please refer to the review papers [11, 26]. We note that we do not consider utilizing an extra clean validation dataset, such as meta-learning-based methods [22, 33] do.

Clean sample selection Most sample selection methods fall into two main categories:

- *Prediction-based methods.* Most of the recent sample selection methods do so, by relying on the predictions of the model classifier, for example on the per-sample loss [2, 14] or model prediction [18, 25]. However, the prediction-based selection is often unstable and easily leads to confirmation bias, especially in heavy noise scenarios. A few works focus on improving the sample selection quality of these methods [31, 40]. To identify open-set noise, several methods utilize the Shannon entropy of the model predictions of different samples [1, 23]. Open-set noise samples that do not belong to any class should have a relatively average model prediction (larger entropy value).
- *Feature-based methods.* Instead of selecting samples based on the model prediction, some works try to utilize the feature representations for sample selection. Wu et al. [29, 30] try to build a KNN graph and identify clean samples through connected subgraphs. Bahri et al. [3] selects clean samples with a KNN classifier in the prediction logit space, while Ortego et al. [20] proposes an iterative KNN to alleviate the effect of noisy labels.

Our work falls in the second category. However, unlike existing methods that use complex variants of neighbouring algorithms, in our pursuit of simplicity and robustness, we use the simplest KNN classification and show that this is sufficient.

Fully exploiting the whole dataset To fully utilise the whole dataset during training and more specifically the non-selected subset, recent methods usually apply semi-supervised training methods (e.g., MixMatch [4]), by considering the selected subset as labelled and the non-selected subset as unlabeled [14]. However, most current semi-supervised learning methods can not deal with open-set samples properly. How to properly do semi-supervised learning in this setting is often referred to as open-set semi-supervised learning [24, 35]. In this paper, instead of adopting complex semi-supervised learning schemes, we adopt a simple relabeling and selection scheme in order to construct a clean and well-labelled subset and then train with a simple cross-entropy loss on the clean, well-labelled set and optionally, with a feature consistency loss on the whole dataset that possibly contains open-set noise and samples that cannot be well relabelled.

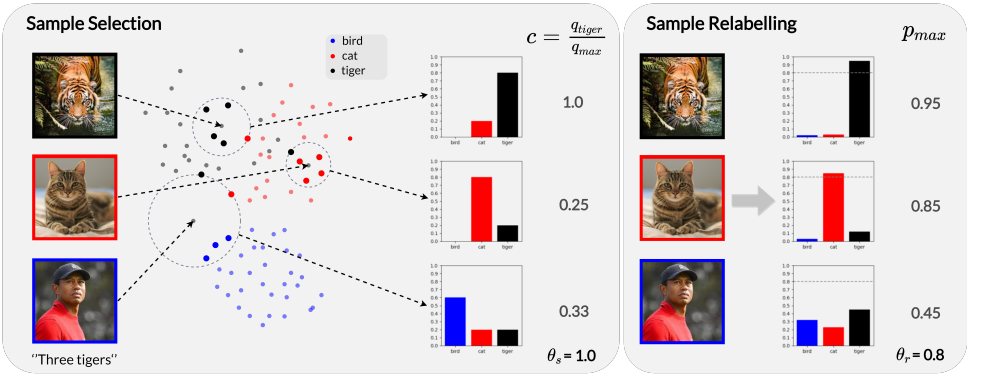

Figure 2: A toy example of **SSR** (section 3.2) with a noisy animal dataset.

3 Methodology

3.1 Problem formulation

Let us denote with $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^N, \boldsymbol{x}_i \in R^d$, a training set with the corresponding one-hot vector labels $\mathcal{Y} = \{\boldsymbol{y}_i\}_{i=1}^N, \boldsymbol{y}_i \in \{0,1\}^M$, where M is the number of classes and N is the number of samples. For convenience, let us also denote the label of each sample $\boldsymbol{x}_i$ corresponding to the one-hot label vector $\boldsymbol{y}_i$ as $l_i = \arg_j[y_i(j) = 1] \in \{1,...,M\}$. Finally, let us denote the true labels with $\mathcal{Y}' = \{\boldsymbol{y}'_i\}_{i=1}^N$. Clearly, for an open-set noisy label it is the case that $\boldsymbol{y}'_i \neq \boldsymbol{y}_i, \boldsymbol{y}'_i \notin \{0,1\}^M$, while for closed-set noisy samples $\boldsymbol{y}'_i \neq \boldsymbol{y}_i, \boldsymbol{y}'_i \in \{0,1\}^M$.

We view the classification network as an encoder f that extracts a feature representation and a **p**arametric **m**odel **c**lassifier (PMC) g_p that deals with the classification problem in question. We also define a **n**on-**p**arametric **K**NN classifier (NPK) g_q based on the feature representations from encoder f. For brevity, we define $\boldsymbol{f}_i \triangleq f(\boldsymbol{x}_i)$ as the feature representation of sample $\boldsymbol{x}_i$, and $\boldsymbol{p}_i \triangleq g_p(\boldsymbol{f}_i)$ and $\boldsymbol{q}_i \triangleq g_q(\boldsymbol{f}_i)$ as the prediction vectors from PMC g_p and NPK g_q, respectively. Following recent works [10, 14, 20, 30, 36], we adopt an iterative scheme for our method consisting of two stages: 1. sample selection (line 3) and relabelling (line 2), and 2. model training (line 4) in **Algorithm 1**.

Algorithm 1: SSR.

Input: dataset $(\mathcal{X}, \mathcal{Y})$, sample selection threshold θ_s, sample relabelling threshold θ_r, weight of feature consistency loss λ, max epochs T

1 **while** $i < T$ **do**
2 Generate $(\mathcal{X}, \mathcal{Y}^r)$ with eq. (3) ; `/* sample relabelling */`
3 Generate $(\mathcal{X}_c, \mathcal{Y}_c^r)$ with eq. (1) and eq. (2) ; `/* sample selection */`
4 Model training with eq. (5) ; `/* model training */`
5 **end**

3.2 Sample selection and relabelling

For a better exposition, we first introduce our sample selection mechanism. Please note, that we actually relabel the samples before each sample selection.

Clean sample selection by balanced neighbouring voting Our sample selection is based on the consistency, as quantified by a measure c_i, between the label $\boldsymbol{y}_i^r$ [1] of sample $\boldsymbol{x}_i$ and an (adjusted) distribution, $\boldsymbol{q}_i$, of the labels in its neighbourhood in the feature space. More specifically, let us denote the similarity between the representations $\boldsymbol{f}_i$ and $\boldsymbol{f}_j$ of any two samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ by $\boldsymbol{s}_{ij}, i, j = 1, ..., N$. By default, we used the cosine similarity, that is, $\boldsymbol{s}_{ij} \triangleq \frac{\boldsymbol{f}_i^T \boldsymbol{f}_j}{\|\boldsymbol{f}_i\|_2 \|\boldsymbol{f}_j\|_2}$. Let us also denote by N_i the index set of the K nearest neighbours of sample $\boldsymbol{x}_i$ in $\mathcal{X}$ based on the calculated similarity. Then, for each sample $\boldsymbol{x}_i$, we can calculate the KNN-voted label distribution $\boldsymbol{q}_i' = \frac{1}{K} \sum_{n \in N_i} \boldsymbol{y}_n^r$ in its neighbourhood, and a balanced version, $\boldsymbol{q}_i \in R^M$, of it that takes into consideration/compensates for the distribution $\boldsymbol{\pi} = \sum_{i=1}^N \boldsymbol{y}_i^r$ of the labels in the dataset. More specifically,

$$\boldsymbol{q}_i = \boldsymbol{\pi}^{-1} \boldsymbol{q}_i', \tag{1}$$

where we denote with $\boldsymbol{\pi}^{-1}$ the vector whose entries are the inverses of the entries of the vector $\boldsymbol{\pi}$ — in this way we alleviate the negative impact of possible class imbalances in sample selection.

The vector $\boldsymbol{q}_i$ can be considered as the (soft) prediction of the NPK g_q classifier. We then, define a consistency measure c_i between the sample's label $l_i^r = \arg\max_j \boldsymbol{y}_i^r(j)$ (section 3.1) and the prediction $\boldsymbol{q}_i$ of the NPK as

$$c_i = \frac{\boldsymbol{q}_i(l_i^r)}{\max_j \boldsymbol{q}_i(j)}, \tag{2}$$

that is the ratio of the value of the distribution $\boldsymbol{q}_i$ at the label l_i^r (eq. (3)) divided by the value of its highest peak $\max_j \boldsymbol{q}_i(j)$. Roughly speaking, a high consistency measure c_i at a sample $\boldsymbol{x}_i$ means that its neighbours agree with its current label l_i^r — this indicates that l_i^r is likely to be correct. By setting a threshold θ_s to c_i, a clean subset $(\mathcal{X}_c, \mathcal{Y}_c^r)$ can be extracted. In our method, we set $\theta_s = 1$ by default, that is, we consider a sample $\boldsymbol{x}_i$ to be clean only when its neighbours' voting $\boldsymbol{q}_i$ is consistent with its current label $\boldsymbol{y}_i^r$.

Noisy sample relabelling by classifier thresholding Our sample relabelling scheme aims at adding well-labelled samples to the training pool and is based on the PMC classifier g_p. Specifically, we "relabel" all samples for which the classifier is confident, that is all samples i for which the prediction $\boldsymbol{p}_i$ of the classifier PMC g_p exceeds a threshold θ_r. Formally,

$$l_i^r = \begin{cases} \arg\max_l \boldsymbol{p}_i(l), & \max_l \boldsymbol{p}_i(l) > \theta_r \\ l_i, & \max_l \boldsymbol{p}_i(l) \leq \theta_r \end{cases} \tag{3}$$

Please note, that similarly to section 3.1, we denote the one-hot label corresponding to l_i^r as $\boldsymbol{y}_i^r$ — this will be used in eq. (1). By setting a high θ_r, a highly confident sample $\boldsymbol{x}_i$ will be relabelled — this can in turn further enhance the quality of sample selection. Note, that this scheme typically avoids mis-relabelling open-set noise samples as those tend not to have highly confident predictions. In this way, our method can deal with open-set noise datasets effectively even though we do not explicitly propose a mechanism for them.

[1] Please note, we use the labels $\mathcal{Y}^r$ (eq. (3)) that a relabelling mechanism provides as mentioned above.

On choices of sample selection and relabelling In this work, we utilize two different classifiers for the two stages of our scheme: for sample selection the NPK g_q based on nearest neighbours in the feature space ((eq. (1) and eq. (2)) and for sample relabelling the PMC g_p classifier (eq. (3)). Here, we justify/comment on this choice.

Why not PMC g_p for sample selection? Most previous works rely on the PMC g_p itself to select clean samples, i.e., typically, samples with small losses. However, it is well known that such methods are not robust to complex and heavy noise. Also, these methods often require a warmup stage before sample selection — this requires prior knowledge about the difficulty and the noise ratio of the dataset. For example, a warmup stage of 50 epochs under heavy noise on the CIFAR10 dataset may result in overfitting before sample selection, while a warmup stage of 5 epochs on the mildly noisy CIFAR100 dataset may not be enough. In this paper, we rely on the smoothness of feature representations and use the NPK g_q for sample selection – even a randomly initialized encoder can provide quite meaningful neighbouring relations therefore enabling us to train the model from scratch and also improves the sample selection stability and performance in heavy and complex noisy scenarios. For a more detailed discussion, please refer to `Supplementary C`.

Why not NPK g_q for sample relabelling? Due to the existence of noisy labels, we found that it is very difficult to make a proper choice of θ_r and rely on the NPK g_q for sample relabeling, especially in the early iterations. By contrast, in our scheme, PMC g_p is always trained with a relatively clean subset and can perform sample relabeling more accurately. Furthermore, relabeling samples on which the classifier is confident leads to smaller gradients and smoother learning from easier samples first — even when the newly assigned labels are wrong, the influence is smaller.

3.3 Model training

In the training stage, we use the most basic form of supervised learning, i.e., using the cross-entropy loss on the clean subset selected in the first stage — this updates both the encoder f and the PMC g_p. With our sample relabelling mechanism, the size of the clean subset grows progressively by including more and more relabeled closed-set noise in training. Optionally, we use a feature consistency loss that enforces consistency between the feature representations of different augmentations of the same sample — this updates the encoder f and helps to learn a strong feature space on which the selection mechanism of the first stage can rely.

Supervised training using the clean subset For each sample $(\boldsymbol{x}, \boldsymbol{y}^r)$ in the selected subset $(\mathcal{X}_c,\ \mathcal{Y}_c^r)$, we train the encoder f and PMC g_p with common cross-entropy loss, that is, $L_{ce} = -\boldsymbol{y}^{rT} \log g_p(f(\boldsymbol{x}))$. Moreover, to deal with the possible class imbalance in the selected subset, we simply over-sample minority classes. In the ablations study, we report the effect of balancing – the over-sampling and also the balanced sample selection in eq. (1).

Optional: feature consistency regularization using all samples Although our relabeling method can progressively relabel and introduce closed-set noise samples into training, open-set samples can also improve generalization. Motivated by commonly used prediction consistency regularization methods, we propose a feature consistency loss L_{fc} [5]. Specifically, with a projector h_{proj} and predictor h_{pred}, we minimize the cosine distance[2] between

[2] In `Supplementary D` we investigate the use of the L2 distance.

two different augmented views ($\boldsymbol{x}_1$ and $\boldsymbol{x}_2$) of the same sample $\boldsymbol{x}$. That is,

$$L_{fc} = -\frac{\boldsymbol{h}_1^\top \boldsymbol{h}_2}{\|\boldsymbol{h}_1\|_2 \|\boldsymbol{h}_2\|_2}, \tag{4}$$

where $\boldsymbol{h}_1 \triangleq h_{pred}(h_{proj}(f(\boldsymbol{x}_1)))$ and $\boldsymbol{h}_2 \triangleq h_{proj}(f(\boldsymbol{x}_2))$.In summary, the overall training objective is to minimize a weighted sum of L_{ce} and L_{fc}, that is

$$L = L_{ce} + \lambda L_{fc}. \tag{5}$$

We set $\lambda = 1$. For brevity, we name our method as SSR when $\lambda = 0$, and SSR+ when $\lambda \neq 0$.

4 Experiments

4.1 Overview

In this section, we conduct extensive experiments on two standard benchmarks with synthetic label noise, CIFAR-10 and CIFAR-100, and three real-world datasets, Clothing1M [32], WebVision [16], and ANIMAL-10N [25]. For brevity, we define abbreviated names for the corresponding noise settings, such as "sym50" for 50% symmetric noise, "asym40" for 40% asymmetric noise and "all30_open50" for 30% total noise with 50% open-set noise. (more dataset and implementation details can be found in `Supplementary A and B`). In section 4.2, we conduct extensive ablation experiments to show the great performance and robustness of our sample selection and relabelling mechanism w.r.t its hyperparameters with different noise types, noise ratios and dataset. In section 4.3 and section 4.4, we compare our method with the state-of-the-art in synthetic noisy datasets and real-world noisy datasets.

4.2 Ablations study

Quality of sample selection and relabelling In fig. 3, we investigate the quality of sample selection and relabeling under different noise types and ratios on the CIFAR10 noisy dataset. We set $\theta_r = 0.9$ for 40asym, 20sym, all30_open50 and all30_open100 noise, while $\theta_r = 0.8$ for sym50 and sym90 noise – please refer to `Supplementary A` for more details on noise. We set $\theta_s = 1$ in all experiments.

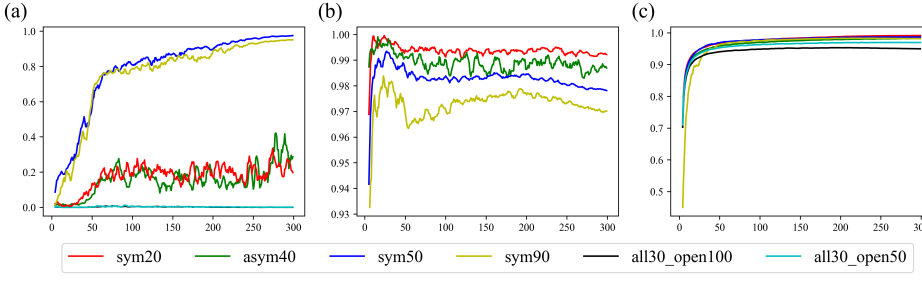

Figure 3: Effect of our sample selection and relabelling method with various noise settings. (a) The proportion of relabeled samples; (c) The corrected clean samples ratio within the relabeled part; (c)F-score of sample selection.

To summarise, we found that the number of the relabeled samples is highly related to the value of θ_r across different noise ratios (fig. 3(a)) for closed-set noise only dataset, for e.g, a lower θ_r leads to more relabeled samples across different noise settings (40asym, 20sym, 50sym and 90sym). For datasets containing also open-set noise (all30_open50 and all30_open100), our relabeling mechanism is more conservative (nearly no relabelling), thus alleviating the negative impact of open-set noise. In fig. 3(b), we show that we obtain very high relabelling accuracy with different noise and relabelling volumes, e.g., only 19% samples have correct labels originally for 90% symmetric noise while >95% samples are correctly relabelled. In fig. 3(c), we report the F-score of our sample selection. Please note that our sample selection is more challenging compared to previous sample selection methods. While previous methods usually focus on identifying clean subsets and noisy subsets based on the original labels, our method involves the relabeling of samples, which takes the risk of introducing more errors while increasing the number of clean subsets. Even so, we see that the F-score of our sample selection works well (> 0.95 in most cases).

Robustness to hyper-parameters In this section, we conduct extensive ablation studies to show the robustness of the values of the few hyperparameters with different noise types, noise ratios, and datasets. The choice of θ_r controls the sample relabelling quality and proportion. Roughly speaking, the lower the θ_r, the more samples will be relabelled. Similarly, the choice of θ_s controls the sample selection quality and proportion – the lower the θ_s, the more samples will be selected for the training stage. We set $\theta_s = 1$ for θ_r ablations and $\theta_r = 0.9$ for θ_s ablations. We also investigate the effects of K — the size of neighborhood of NPK g_q during the sample selection stage, with $\theta_s = 1$ and $\theta_r = 0.9$.

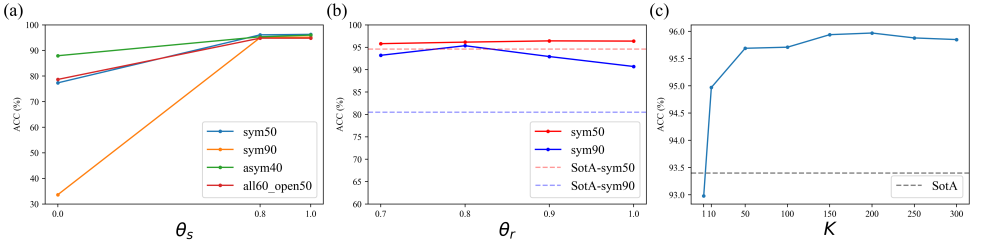

Figure 4: Classification accuracy with different hyper-parameters on CIFAR10 datasets. (a)$\theta_s = [0, 0.8, 1.0]$; (b)$\theta_r = [0.7, 0.8, 0.9, 1]$; (c) $K = [1, 10, 50, 100, 150, 200, 250, 300]$.

In fig. 4(a) we report results with $\theta_s = [0, 0.8, 1.0]$. Removing sample selection ($\theta_s = 0$) leads to severe degradation especially for a high noise ratio (90% symmetric noise), while a relatively high θ_s gives consistently high performance. In fig. 4(b), we report the performance with different θ_r on the synthetic CIFAR10 noisy dataset. Our method achieves consistently superior performance than the state-of-the-art with different θ_r. In fig. 4(c), we report results with different K for the CIFAR10 dataset with 40% asymmetric noise since it is more challenging and realistic. Except for extremely small K, our method is stable and consistently better than the state-of-the-art.

Effect of balancing strategies To alleviate the possible class imbalance in the dataset, we proposed two balancing strategies, one in the sample selection [eq. (1)] and one in the model training stage [Over-sampling minority class], respectively. In table 1 we investigate the

effect of using data balancing or not, on CIFAR10 with 40% asymmetric noise and also on a well-known real-world imbalanced noisy dataset, Clothing1M. It can be seen that the effect is small but positive.

Method	Clothing1M	40% asym CIFAR10
SSR	**74.83**	**95.5**
w/o balancing	74.12	94.9

Table 1: Effect of class balancing.

4.3 Evaluation with synthetic noisy datasets

In this section, we compare our method to the most recent state-of-the-art methods and we show that it achieves consistent improvements in all datasets and at all noise types and ratios.

Dataset	CIFAR10					CIFAR100			
Noise type	Symmetric				Assymetric	Symmetric			
Noise ratio	20%	50%	80%	90%	40%	20%	50%	80%	90%
Cross-Entropy	86.8	79.4	62.9	42.7	85.0	62.0	46.7	19.9	10.1
Co-teaching+ [38]	89.5	85.7	67.4	47.9	-	65.6	51.8	27.9	13.7
F-correction [21]	86.8	79.8	63.3	42.9	87.2	61.5	46.6	19.9	10.2
PENCIL [34]	92.4	89.1	77.5	58.9	88.5	69.4	57.5	31.1	15.3
LossModelling [1]	94.0	92.0	86.8	69.1	87.4	73.9	66.1	48.2	24.3
DivideMix* [14]	96.1	94.6	93.2	76.0	93.4	77.3	74.6	60.2	31.5
ELR+* [17]	95.8	94.8	93.3	78.7	93.0	77.6	73.6	60.8	33.4
RRL [15]	95.8	94.3	92.4	75.0	91.9	79.1	74.8	57.7	29.3
NGC [30]	95.9	94.5	91.6	80.5	90.6	79.3	75.9	62.7	29.8
AugDesc* [19]	96.3	95.4	93.8	91.9	94.6	79.5	77.2	66.4	41.2
C2D* [35]	96.4	95.3	94.4	93.6	93.5	78.7	76.4	67.8	58.7
SSR(ours)	96.3	95.7	95.2	94.6	95.1	79.0	75.9	69.5	61.8
SSR+(ours)	**96.7**	**96.1**	**95.6**	**95.2**	**95.5**	**79.7**	**77.2**	**71.9**	**66.6**

Table 2: Evaluation on CIFAR-10 and CIFAR-100 with closed-set noise. Methods marked with an asterisk employ semi-supervised learning, model co-training or model pre-training.

Evaluation with controlled closed-set noise In this section, we compare SSR/SSR+ to the most competitive recent works. Table 2 shows results on CIFAR10 and CIFAR100 — we note again for SSR/SSR+ this is without the use of model cotraining or pre-training. It is clear that our method far outperforms them (e.g. 66.6% accuracy on CIFAR100 with 90% symmetric noise), not only in the case of symmetric noise but also in the more realistic asymmetric synthetic noise settings.

Evaluation with combined open-set noise and closed-set noise Table 4 shows the performance of our method in a more complex combined noise scenario. Previous methods that are specially designed for open-set noise degrade rapidly when the open-set noise ratio is decreased from 1 to 0.5 [13, 27]. Also, the performance of the method without considering open-set noise like DivideMix [14] decreases when the open-set noise ratio is increased. EDM [23] modifies the method of DivideMix to deal with combined noise, however, reports results that are considerably lower than ours.

CE	F-correction [23]	ELR [15]	RRL [13]	C2D* [33]	DivideMix* [17]	ELR+* [15]	AugDesc* [6]	SSR+(ours)
69.21	69.84	72.87	74.30	74.84	74.76	74.81	**75.11**	**74.83**

Table 3: Testing accuracy (%) on Clothing1M (methods with * utilized model cotraining).

Method	Noise ratio	0.3		0.6	
	Open ratio	0.5	1	0.5	1
ILON [22]	Best	87.4	90.4	80.5	83.4
	Last	80.0	87.4	55.2	78.0
RoG [35]	Best	89.8	91.4	84.1	88.2
	Last	85.9	89.8	66.3	82.1
DivideMix [17]	Best	91.5	89.3	91.8	89.0
	Last	90.9	88.7	91.5	88.7
EDM [25]	Best	94.5	92.9	93.4	90.6
	Last	94.0	91.9	92.8	89.4
SSR(ours)	Best	96.0	95.7	93.8	93.1
	Last	95.9	95.6	93.7	93.1
SSR+(ours)	Best	**96.3**	**96.1**	**95.2**	**94.0**
	Last	**96.2**	**96.0**	**95.2**	**93.9**

Table 4: Evaluation on CIFAR10 with combined noise.

Methods	WebVision		ILSVRC2012	
	Top1	Top5	Top1	Top5
Co-teaching [11]	63.58	85.20	61.48	84.70
DivideMix [17]	77.32	91.64	75.20	90.84
ELR+ [15]	77.78	91.68	70.29	89.76
NGC [30]	79.16	91.84	74.44	91.04
LongReMix [1]	78.92	92.32	-	-
RRL [13]	76.3	91.5	73.3	91.2
SSR+(ours)	**80.92**	**92.80**	**75.76**	**91.76**

Table 5: Testing accuracy (%) on Webvision.

Cross-Entropy	SELFIE [25]	PLC [37]	NCT [1]	SSR+(ours)
79.4	81.8	83.4	84.1	**88.5**

Table 6: Testing accuracy on ANIMAL-10N.

4.4 Evaluation with real-world noisy datasets

Finally, in table 3, table 5 and table 6 we show results on the Clothing1M, WebVision and ANIMAL-10N datasets, respectively. To summarize, our method achieves better or competitive performance in relation to the current state-of-the-art in both large-scale web-crawled datasets and small-scale human-annotated noisy datasets.

5 Conclusions

In this paper we propose an efficient ***S**ample **S**election and **R**elabelling* (SSR) framework for ***L**earning with **U**nknown **L**abel **N**oise* (**LULN**). Unlike previous methods that try to integrate many different mechanisms and regularizations, we strive for a concise, simple and robust method. The proposed method does not utilize complicated mechanisms such as semi-supervised learning, model co-training and model pre-training, and is shown with extensive experiments and ablation studies to be robust to the values of its few hyper-parameters, and to consistently and by large surpass the state-of-the-art in various datasets.

Acknowledgments: This work was supported by the EU H2020 AI4Media No. 951911 project.

References

[1] Paul Albert, Diego Ortego, Eric Arazo, Noel E O'Connor, and Kevin McGuinness. Addressing out-of-distribution label noise in webly-labelled data. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 392–401, 2022.

[2] Eric Arazo, Diego Ortego, Paul Albert, Noel O'Connor, and Kevin McGuinness. Unsupervised label noise modeling and loss correction. In *International Conference on Machine Learning*, pages 312–321. PMLR, 2019.

[3] Dara Bahri, Heinrich Jiang, and Maya Gupta. Deep k-nn for noisy labels. In *International Conference on Machine Learning*, pages 540–550. PMLR, 2020.

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

[5] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758, 2021.

[6] Yingyi Chen, Xi Shen, Shell Xu Hu, and Johan AK Suykens. Boosting co-teaching with compression regularization for label noise. *arXiv preprint arXiv:2104.13766*, 2021.

[7] Filipe R Cordeiro, Ragav Sachdeva, Vasileios Belagiannis, Ian Reid, and Gustavo Carneiro. Longremix: Robust learning with high confidence samples in a noisy label environment. *arXiv preprint arXiv:2103.04173*, 2021.

[8] Aritra Ghosh, Himanshu Kumar, and PS Sastry. Robust loss functions under label noise for deep neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

[9] Jacob Goldberger and Ehud Ben-Reuven. Training deep neural-networks using a noise adaptation layer. 2016.

[10] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *arXiv preprint arXiv:1804.06872*, 2018.

[11] Bo Han, Quanming Yao, Tongliang Liu, Gang Niu, Ivor W Tsang, James T Kwok, and Masashi Sugiyama. A survey of label-noise representation learning: Past, present and future. *arXiv preprint arXiv:2011.04406*, 2020.

[12] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International Conference on Machine Learning*, pages 2304–2313. PMLR, 2018.

[13] Kimin Lee, Sukmin Yun, Kibok Lee, Honglak Lee, Bo Li, and Jinwoo Shin. Robust inference via generative classifiers for handling noisy labels. In *International Conference on Machine Learning*, pages 3763–3772. PMLR, 2019.

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

[15] Junnan Li, Caiming Xiong, and Steven Hoi. Learning from noisy data with robust representation learning. 2020.

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

[17] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *arXiv preprint arXiv:2007.00151*, 2020.

[18] Eran Malach and Shai Shalev-Shwartz. Decoupling" when to update" from" how to update". *arXiv preprint arXiv:1706.02613*, 2017.

[19] Kento Nishi, Yi Ding, Alex Rich, and Tobias Hollerer. Augmentation strategies for learning with noisy labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8022–8031, 2021.

[20] Diego Ortego, Eric Arazo, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Multi-objective interpolation training for robustness to label noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6606–6615, 2021.

[21] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1944–1952, 2017.

[22] Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In *International conference on machine learning*, pages 4334–4343. PMLR, 2018.

[23] Ragav Sachdeva, Filipe R Cordeiro, Vasileios Belagiannis, Ian Reid, and Gustavo 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*, pages 3607–3615, 2021.

[24] Kuniaki Saito, Donghyun Kim, and Kate Saenko. Openmatch: Open-set consistency regularization for semi-supervised learning with outliers. *arXiv preprint arXiv:2105.14148*, 2021.

[25] Hwanjun Song, Minseok Kim, and Jae-Gil Lee. Selfie: Refurbishing unclean samples for robust deep learning. In *International Conference on Machine Learning*, pages 5907–5915. PMLR, 2019.

[26] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[27] Yisen Wang, Weiyang Liu, Xingjun Ma, James Bailey, Hongyuan Zha, Le Song, and Shu-Tao Xia. Iterative learning with open-set noisy labels. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8688–8696, 2018.

[28] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 322–330, 2019.

[29] Pengxiang Wu, Songzhu Zheng, Mayank Goswami, Dimitris Metaxas, and Chao Chen. A topological filter for learning with label noise. *Advances in neural information processing systems*, 33:21382–21393, 2020.

[30] Zhi-Fan Wu, Tong Wei, Jianwen Jiang, Chaojie Mao, Mingqian Tang, and Yu-Feng Li. Ngc: A unified framework for learning with open-world noisy data. *arXiv preprint arXiv:2108.11035*, 2021.

[31] Xiaobo Xia, Tongliang Liu, Bo Han, Mingming Gong, Jun Yu, Gang Niu, and Masashi Sugiyama. Sample selection with uncertainty of losses for learning with noisy labels. *arXiv preprint arXiv:2106.00445*, 2021.

[32] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2691–2699, 2015.

[33] Youjiang Xu, Linchao Zhu, Lu Jiang, and Yi Yang. Faster meta update strategy for noise-robust deep learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 144–153, 2021.

[34] Kun Yi and Jianxin Wu. Probabilistic end-to-end noise correction for learning with noisy labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7017–7025, 2019.

[35] Qing Yu, Daiki Ikami, Go Irie, and Kiyoharu Aizawa. Multi-task curriculum framework for open-set semi-supervised learning. In *European Conference on Computer Vision*, pages 438–454. Springer, 2020.

[36] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? In *International Conference on Machine Learning*, pages 7164–7173. PMLR, 2019.

[37] Yikai Zhang, Songzhu Zheng, Pengxiang Wu, Mayank Goswami, and Chao Chen. Learning with feature-dependent label noise: A progressive approach. *arXiv preprint arXiv:2103.07756*, 2021.

[38] Zhilu Zhang and Mert R Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. *arXiv preprint arXiv:1805.07836*, 2018.

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

[40] Tianyi Zhou, Shengjie Wang, and Jeff Bilmes. Robust curriculum learning: from clean label detection to noisy label self-correction. In *International Conference on Learning Representations*, 2020.