Co-assistant Networks for Label Correction

Xuan Chen, Weiheng Fu, Tian Li, Xiaoshuang Shi(✉), Hengtao Shen, and Xiaofeng Zhu

School of Computer Science and Engineering,
University of Electronic Science and Technology of China, Chengdu 611731, China
xsshi2013@gmail.com

Abstract. The presence of corrupted labels is a common problem in the medical image datasets due to the difficulty of annotation. Meanwhile, corrupted labels might significantly deteriorate the performance of deep neural networks (DNNs), which have been widely applied to medical image analysis. To alleviate this issue, in this paper, we propose a novel framework, namely Co-assistant Networks for Label Correction (CNLC), to simultaneously detect and correct corrupted labels. Specifically, the proposed framework consists of two modules, *i.e.,* noise detector and noise cleaner. The noise detector designs a CNN-based model to distinguish corrupted labels from all samples, while the noise cleaner investigates class-based GCNs to correct the detected corrupted labels. Moreover, we design a new bi-level optimization algorithm to optimize our proposed objective function. Extensive experiments on three popular medical image datasets demonstrate the superior performance of our framework over recent state-of-the-art methods. Source codes of the proposed method are available on https://github.com/shannak-chen/CNLC.

Keywords: Corrupted labels · Label correction · CNN · GCN

1 Introduction

The success of deep neural networks (DNNs) mainly depends on the large number of samples and the high-quality labels. However, either of them is very difficult to be obtained for conducting medical image analysis with DNNs. In particular, obtaining high-quality labels needs professional experience so that corrupted labels can often be found in medical datasets, which can seriously degrade the effectiveness of medical image analysis. Moreover, sample annotation needs expensive cost. Hence, correcting corrupted labels might be one of effective solutions to solve the issues of high-quality labels.

Numerous works have been proposed to tackle the issue of corrupted labels. Based on whether correcting corrupted labels, previous methods can be roughly

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-43898-1_16.

H. Greenspan et al. (Eds.): MICCAI 2023, LNCS 14222, pp. 159–168, 2023.
https://doi.org/10.1007/978-3-031-43898-1_16

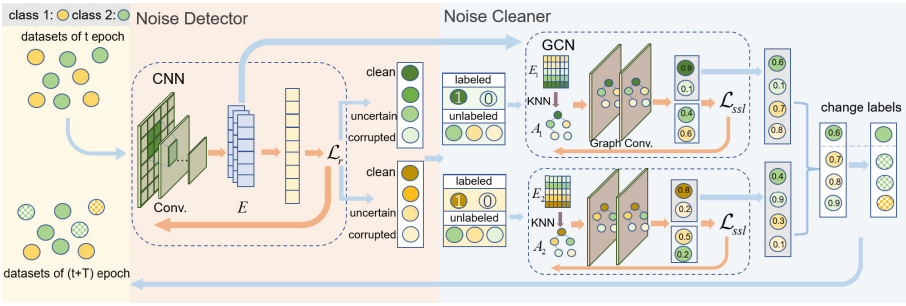


Fig. 1. The architecture of the proposed CNLC framework consists of two modules, *i.e.,* noise detector and noise cleaner. The noise detector outputs the embedding of all training samples and classifies the training samples of each class into three subgroups, including clean samples, uncertain samples and corrupted samples. The noise cleaner constructs a GCN for each class to correct the labels of both corrupted samples and a subset of uncertain samples for all classes.

divided into two categories, *i.e.,* robustness-based methods [12,17] and label correction methods [14,22]. Robustness-based methods are designed to utilize various techniques, such as dropout, augmentation and loss regularization, to avoid the adverse impact of corrupted labels, thereby outputting a robust model. Label correction methods are proposed to first detect corrupted labels and then correct them. For example, co-correction [9] simultaneously trains two models and corrects labels for medical image analysis, and LCC [5] first regards the outputs of DNN as the class probability of the training samples and then changes the labels of samples with low class probability. Label correction methods are significant for disease diagnosis, because physicians can double check the probably mislabeled samples to improve diagnosis accuracy. However, current label correction methods still have limitations to be addressed. First, they cannot detect and correct all corrupted labels, and meanwhile they usually fail to consider boosting the robustness of the model itself, so that the effectiveness of DNNs is possibly degraded. Second, existing label correction methods often ignore to take into account the relationship among the samples so that influencing the effectiveness of label correction.

To address the aforementioned issues, in this paper, we propose a new co-assistant framework, namely Co-assistant Networks for Label Correction (CNLC) (shown in Fig. 1), which consists of two modules, *i.e.,* noise detector and noise cleaner. Specifically, the noise detector first adopts a convolutional neural network (CNN [6,20]) to predict the class probability of samples, and then the loss is used to partition all the training samples for each class into three subgroups, *i.e.,* clean samples, uncertain samples and corrupted samples. Moreover, we design a robust loss (*i.e.,* a resistance loss) into the CNN framework to avoid model overfitting on corrupted labels and thus exploring the first issue in previous label correction methods. The noise cleaner constructs a graph convolutional network (GCN [18,19]) model for each class to correct the corrupted labels. During the process of noise cleaner, we consider the relationship

among samples (*i.e.,* the local topology structure preservation by GCN) to touch the second issue in previous methods. In particular, our proposed CNLC iteratively updates the noise detector and the noise cleaner, which results in a bi-level optimization problem [4,10]

Compared to previous methods, the contributions of our method is two-fold. First, we propose a new label correction method (*i.e.,* a co-assistant framework) to boost the model robustness for medical image analysis by two sequential modules. Either of them adaptively adjusts the other, and thus guaranteeing to output a robust label correction model. Second, two sequential modules in our framework results in a bi-level optimization problem. We thus design a bi-level optimization algorithm to solve our proposed objective function.

2 Methodology

In this section, our proposed method first designs a noise detector to discriminate corrupted samples from all samples, and then investigates a noise cleaner to correct the detected corrupted labels.

2.1 Noise Detector

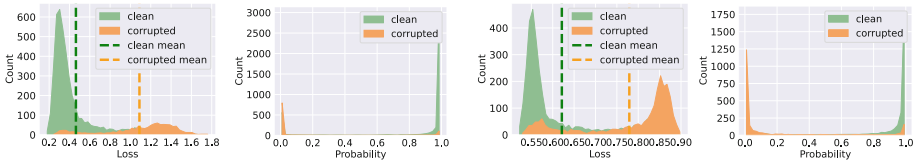
Noise detector is used to distinguish corrupted samples from clean samples. The prevailing detection method is designed to first calculate the loss of DNNs on all training samples and then distinguish corrupted samples from clean ones based on their losses. Specifically, the samples with small losses are regarded as clean samples while the samples with large losses are regarded as corrupted samples.

Different from previous literature [6,7], our noise detector involves two steps, *i.e.,* CNN and label partition, to partition all training samples for each class into three subgroups, *i.e.,* clean samples, uncertain samples and corrupted samples. Specifically, we first employ CNN with the cross-entropy loss as the backbone to obtain the loss of all training samples. Since the cross-entropy loss is easy to overfit on corrupted labels without extra noise-tolerant term [1,21], we change it to the following resistant loss in CNN:

$$\mathcal{L}_r = \frac{1}{b}\sum_{i=1}^{b} -\log\left(p_i^t\left[\tilde{y_i}\right]\right) + \frac{\lambda\left(t\right)}{b}\sum_{i=1}^{b}\sum_{j=1}^{C} -p_i^t\left[j\right]\log p_i^{t-1}\left[j\right], \tag{1}$$

where b is the number of samples in each batch, $p_i^t\left[j\right]$ represents the j-th class prediction of the i-th sample in the t-th epoch, $\tilde{y_i} \in \{0, 1, ..., C-1\}$ denotes the corrupted label of the i-th sample $\mathbf{x}_i$, C denotes the number of classes and $\lambda(t)$ is a time-related hyper-parameter. In Eq. (1), the first term is the cross-entropy loss. The second term is the resistance loss which is proposed to smooth the update of model parameters so that preventing model overfitting on corrupted labels to some extent [12].

In label partition, based on the resistant loss in Eq. (1), the training samples for each class are divided into three subgroups, *i.e.,* clean samples, uncertain



(a) $\epsilon = 0.2$ (b) $\epsilon = 0.4$

Fig. 2. Cross-entropy loss distribution and GMM probability on different noise rates ϵ on BreakHis [13]. "clean" denotes the samples with the ground-truth labels, while "corrupted" denotes the samples with the corrupted labels.

samples, and corrupted samples. Specifically, the samples with n_1 smallest losses are regarded as clean samples and the samples with n_1 largest loss values are regarded as corrupted samples, where n_1 is experimentally set as 5.0% of all training sample for each class. The rest of the training samples for each class are regarded as uncertain samples.

In noise detector, our goal is to identify the real clean samples and real corrupted samples, which are corresponded to set as positive samples and negative samples in noise cleaner. If we select a large number of either clean samples or corrupted samples (*e.g.,* larger than 5.0% of all training samples), they may contain false positive samples or false negative samples, so that the effectiveness of the noise cleaner will be influenced. As a result, our noise detector partitions all training samples for each class into three subgroups, including a small proportion of clean samples and corrupted samples, as well as uncertain samples.

2.2 Noise Cleaner

Noise cleaner is designed to correct labels of samples with corrupted labels. Recent works often employ DNNs (such as CNN [8] and MLP [15]) to correct the corrupted labels. First, these methods ignore to take into account the relationship among the samples, such as local topology structure preservation, *i.e.,* one of popular techniques in computer vision and machine learning, which ensures that nearby samples have similar labels and dissimilar samples have different labels. In particular, based on the partition mentioned in the above section, the clean samples within the same class should have the same label and the corrupted samples should have different labels from clean samples within the same class. This indicates that it is necessary to preserve the local topology structure of samples within the same class. Second, in noise detector, we only select a small proportion of clean samples and corrupted samples for the construction of noise cleaner. Limited number of samples cannot guarantee to build robust noise cleaner. In this paper, we address the above issues by employing semi-supervised learning, *i.e.,* a GCN for each class, which keeps the local topology structure of samples on both labeled samples and unlabeled samples. Specifically, our noise cleaner includes three components, *i.e.,* noise rate estimation, class-based GCNs, and corrupted label correction.

The inputs of each class-based GCN include labeled samples and unlabeled samples. The labeled samples consist of positive samples (*i.e.,* the clean samples of this class with the new label $z_{ic} = 1$ for the i-th sample in the c-th class) and negative samples (*i.e.,* the corrupted samples of this class with the new label $z_{ic} = 0$). The unlabeled samples include a subset of the uncertain samples from all classes and corrupted samples of other classes. We follow the principle to select uncertain samples for each class, *i.e.,* the higher the resistant loss in Eq. (1), the higher the probability of the sample belonging to corrupted samples. Moreover, the number of uncertain samples is determined by noise rate estimation.

Given the resistant loss in Eq. (1), in noise rate estimation, we estimate the noise rate of the training samples by employing a Gaussian mixed model (GMM) composed of two Gaussian models. As shown in Fig. 2, we observe that the mean value of Gaussian model for corrupted samples is greater than that of Gaussian model for clean samples. Thus, the Gaussian model with a large mean value is probably the curve of corrupted labels. Based on this, given two outputs of the GMM model for the i-th sample, its output with a larger mean value and the output with a smaller mean value, respectively, are denoted as $M_{i,1}$ and $M_{i,2}$, so the following definition v_i is used to determine if the i-th samples is noise:

$$v_i = \begin{cases} 1, & M_{i,1} > M_{i,2} \\ 0, & \text{otherwise} \end{cases}. \tag{2}$$

Hence, the noise rate r of training samples is calculated by:

$$r = \frac{\sum_{i=1}^{n} v_i}{n}, \tag{3}$$

where n represents the total number of samples in training dataset. Supposing the number of samples in the c-th class is s_c, the number of uncertain samples of each class is $s_c \times r - n_1$. Hence, the total number of unlabeled samples for each class is $n \times r - n_1$ in noise cleaner.

Given $2 \times n_1$ labeled samples and $n \times r - n_1$ unlabeled samples, the class-based GCN for each class conducts semi-supervised learning to predict $n \times r$ samples, including $n \times r - n_1$ unlabeled samples and n_1 corrupted samples for this class. The semi-supervised loss $\mathcal{L}_{ssl}$ includes a binary cross-entropy loss $\mathcal{L}_{bce}$ for labeled samples and an unsupervised loss $\mathcal{L}_{mse}$ [8] for unlabeled samples, *i.e.,* $\mathcal{L}_{ssl} = \mathcal{L}_{bce} + \mathcal{L}_{mse}$, where $\mathcal{L}_{bce}$ and $\mathcal{L}_{mse}$ are defined as:

$$\mathcal{L}_{bce} = \frac{-1}{2n_1} \sum_{i=1}^{2n_1} z_{ic} log q_{ic}^t + (1 - z_{ic})\, log\left(1 - q_{ic}^t\right), \tag{4}$$

$$\mathcal{L}_{mse} = \frac{1}{n \times r - n_1} \sum_{i=2n_1+1}^{n \times r + n_1} \left\| q_{ic}^t - \check{q}_{ic}^{t-1} \right\|^2, \tag{5}$$

where q_{ic}^t denotes the prediction of the i-th sample in the t-th epoch for the class c, $\check{q}_{ic}^t$ is updated by $\check{q}_{ic}^t = \frac{\rho \times \check{q}_{ic}^{t-1} + (1-\rho) \times q_{ic}^{t-1}}{\varpi(t)}$, where $\varpi(t)$ is related to time [8].

In corrupted label correction, given C well-trained GCNs and the similarity scores on each class for a subset of uncertain samples and all corrupted samples, their labels can be determined by:

$$\tilde{y_i} = \underset{0 \leq c \leq C-1}{\operatorname{argmax}} \left(q_{ic}\right). \tag{6}$$

2.3 Objective Function

The optimization of the noise detector is associated with the corrupted label set $\tilde{\mathbf{y}}$, which is determined by noise cleaner. Similarly, the embedding of all samples $\mathbf{E}$ is an essential input of the noise cleaner, which is generated by the noise detector. As the optimizations of two modules are nested, the objective function of our proposed method is the following bi-level optimization problem:

$$\begin{cases} \min_{\theta} \mathcal{L}_r \left(f^t\left(\mathbf{x};\theta\right), f^{t-1}\left(\mathbf{x};\theta\right), \tilde{\mathbf{y}}\right), \\ \min_{\omega_c} \mathcal{L}_{bce}\left(g_c^t\left(\mathbf{A}_c, \mathbf{E}_c; \omega_c\right), \mathbf{z}_c\right) + \mathcal{L}_{mse}\left(g_c^t\left(\mathbf{A}_c, \mathbf{E}_c; \omega_c\right), g_c^{t-1}\left(\mathbf{A}_c, \mathbf{E}_c; \omega_c\right)\right), \end{cases} \tag{7}$$

where $f^t(\mathbf{x};\theta)$ denotes the output of the upper-level (*i.e.,* the noise detector) in the t-th epoch, $\mathbf{A}_c$ and $\mathbf{E}_c$ represent the adjacency matrix and the feature matrix of class c, $\mathbf{z}_c$ are labels of labeled samples in class c, $g_c^t(\mathbf{A}_c, \mathbf{E}_c; \omega_c)$ and $g_c^{t-1}(\mathbf{A}_c, \mathbf{E}_c; \omega_c)$ denote the output of GCN model for class c at the t-th and t-1-th epochs, respectively.

In this paper, we construct a bi-level optimization algorithm to search optimal network parameters of the above objective function. Specifically, we optimize the noise detector to output an optimal feature matrix $\mathbf{E}^*$, which is used for the construction of the noise cleaner. Furthermore, the output $\tilde{\mathbf{y}}^*$ of the noise cleaner is used to optimize the noise detector. This optimization process alternatively optimize two modules until the noise cleaner converges. We list the optimization details of our proposed algorithm in the supplemental materials.

3 Experiments

3.1 Experimental Settings

The used datasets are **BreakHis** [13], **ISIC** [3], and **NIHCC** [16]. **BreakHis** consists of 7,909 breast cancer histopathological images including 2,480 benigns and 5,429 malignants. We randomly select 5,537 images for training and 2,372 ones for testing. **ISIC** has 12,000 digital skin images where 6,000 are normal and 6,000 are with melanoma. We randomly choose 9,600 samples for training and the remaining ones for testing. **NIHCC** has 10,280 frontal-view X-ray images, where 5,110 are normal and 5,170 are with lung diseases. We randomly select 8,574 images for training and the rest of images for testing. In particular, the random selection in our experiments guarantees that three datasets (*i.e.,* the training set, the testing set, and the whole set) have the same ratio for each

Table 1. The classification results (average ± std) on three datasets.

Dataset	Method	$\epsilon = 0.2$				$\epsilon = 0.4$			
		ACC	SEN	SPE	AUC	ACC	SEN	SPE	AUC
BreakHis	CE	82.7±1.7	87.2±2.4	73.1±3.2	80.1±1.7	64.4±2.4	67.0±2.4	58.9±5.5	62.9±3.0
	CT	87.3±1.0	92.5±5.6	76.1±8.5	84.3±0.6	84.4±0.3	93.8±1.3	63.9±2.0	78.9±0.4
	NCT	87.4±0.1	95.0±0.4	70.8±1.0	82.9±0.3	82.9±0.4	98.0±0.2	49.8±1.7	73.9±0.7
	SPRL	86.1±0.1	95.9±0.5	64.8±3.5	80.4±0.4	82.0±0.1	**99.0**±0.2	44.6±3.2	71.8±1.5
	CC	87.8±0.0	95.9±0.2	70.1±0.4	83.0±0.1	84.1±0.1	97.8±0.1	54.1±0.0	76.0±0.2
	SELC	86.6±0.1	**95.9**±0.1	66.3±2.6	81.1±0.3	82.7±0.1	98.1±0.1	49.0±3.0	73.5±0.5
	CNLC	**90.1**±0.2	95.0±0.4	**79.3**±1.2	**87.1**±0.4	**85.2**±0.1	94.3±0.8	**65.3**±2.0	**79.8**±0.5
ISIC	CE	80.4±1.4	79.8±3.8	81.1±3.7	80.4±1.4	60.1±2.0	58.2±4.5	62.1±4.2	59.5±2.8
	CT	88.1±0.3	88.0±0.6	88.2±0.6	88.1±0.3	84.3±0.3	78.7±0.6	90.0±0.4	84.3±0.3
	NCT	88.3±0.1	86.5±0.5	90.2±0.3	88.4±0.1	82.1±0.3	75.8±1.1	88.7±0.7	85.2±0.3
	SPRL	88.5±0.1	88.5±0.1	88.5±0.3	88.5±0.1	84.1±0.2	83.1±0.4	85.0±0.3	84.1±0.2
	CC	84.5±0.2	82.4±0.5	86.7±0.1	84.5±0.2	83.8±0.1	81.2±0.2	86.5±0.1	83.8±0.1
	SELC	88.1±0.0	86.5±0.5	89.8±0.4	88.1±0.1	79.2±0.4	65.3±1.2	**93.5**±0.3	79.4±0.4
	CNLC	**90.4**±0.2	**89.1**±0.4	**91.8**±0.2	**90.4**±0.2	**85.5**±0.2	**83.2**±0.6	87.9±0.3	**85.5**±0.2
NIHCC	CE	78.4±1.4	70.0±5.2	86.8±3.6	78.4±1.4	66.9±1.7	61.9±6.8	71.8±6.5	66.9±1.7
	CT	82.7±0.3	78.7±0.9	86.6±0.7	82.7±0.3	73.7±0.3	68.8±0.7	78.6±0.7	73.7±0.2
	NCT	81.9±0.1	83.9±0.7	79.9±1.0	81.9±0.1	73.7±0.1	69.8±0.4	77.6±0.2	73.7±0.1
	SPRL	82.3±0.1	77.1±0.3	87.6±0.3	82.4±0.1	74.8±0.1	65.9±0.3	**83.8**±0.2	74.9±0.1
	CC	78.0±0.1	65.5±0.4	**90.5**±0.2	78.0±0.1	67.7±0.1	54.3±0.3	81.2±0.1	67.8±0.1
	SELC	79.6±0.2	78.2±0.4	81.1±0.9	79.6±0.2	71.4±0.1	72.9±0.5	69.9±0.5	71.4±0.1
	CNLC	**84.9**±0.4	**85.0**±1.4	84.8±2.3	**84.9**±0.4	**77.9**±0.2	**73.8**±1.7	82.1±1.4	**78.0**±0.2

class. Moreover, we assume that all labels in the used raw datasets are clean, so we add corrupted labels with different noise rates $\epsilon = \{0, 0.2, 0.4\}$ into these datasests, where $\epsilon = 0$ means that all labels in the training set are clean.

We compare our proposed method with six popular methods, including one fundamental baseline (*i.e.,* Cross-Entropy (CE)), three robustness-based methods (*i.e.,* Co-teaching (CT) [6], Nested Co-teaching (NCT) [2] and Self-Paced Resistance Learning (SPRL) [12]), and two label correction methods (*i.e.,* Co-Correcting (CC) [9] and Self-Ensemble Label Correction (SELC) [11]). For fairness, in our experiments, we adopt the same neural network for all comparison methods based on their public codes and default parameter settings. We evaluate the effectiveness of all methods in terms of four evaluation metrics, *i.e.,* classification accuracy (ACC), specificity (SPE), sensitivity (SEN) and area under the ROC curve (AUC).

3.2 Results and Analysis

Table 1 presents the classification results of all methods on three datasets. Due to the space limitation, we present the results at $\epsilon = 0.0$ of all methods in the supplemental materials. First, our method obtains the best results, followed by CT, NCT, SPRL, CELC, CC, and CE, on all datasets in terms of four evalua-

Table 2. The classification results (average ± std) of the ablation study on ISIC.

	$\epsilon = 0.2$				$\epsilon = 0.4$			
Method	ACC	SEN	SPE	AUC	ACC	SEN	SPE	AUC
W/O NC	$88.5_{\pm 0.2}$	$86.7_{\pm 0.8}$	$89.1_{\pm 1.9}$	$88.1_{\pm 0.5}$	$81.9_{\pm 0.7}$	$78.9_{\pm 2.6}$	$85.0_{\pm 2.3}$	$82.0_{\pm 0.1}$
MLP	$90.0_{\pm 0.2}$	$89.0_{\pm 0.2}$	$91.0_{\pm 0.4}$	$90.0_{\pm 0.2}$	$84.1_{\pm 0.3}$	$77.6_{\pm 0.7}$	$\mathbf{90.7}_{\pm 0.5}$	$84.2_{\pm 0.3}$
CNLC-RL	$89.5_{\pm 0.4}$	$88.4_{\pm 1.0}$	$90.6_{\pm 0.9}$	$89.5_{\pm 0.4}$	$83.5_{\pm 0.4}$	$82.2_{\pm 1.0}$	$84.9_{\pm 0.6}$	$83.5_{\pm 0.4}$
CNLC	$\mathbf{90.4}_{\pm 0.2}$	$\mathbf{89.1}_{\pm 0.4}$	$\mathbf{91.8}_{\pm 0.2}$	$\mathbf{90.4}_{\pm 0.2}$	$\mathbf{85.5}_{\pm 0.2}$	$\mathbf{83.2}_{\pm 0.2}$	$87.9_{\pm 0.3}$	$\mathbf{85.5}_{\pm 0.2}$

tion metrics. For example, our method on average improves by 2.4% and 15.3%, respectively, compared to the best comparison method (*i.e.,* CT) and the worst comparison method (*i.e.,* CE), on all cases. This might be because our proposed method not only utilizes a robust method to train a CNN for distinguishing corrupted labels from clean labels, but also corrects them by considering their relationship among the samples within the same class. Second, all methods outperform the fundamental baseline (*i.e.,* CE) on all cases. For example, the accuracy of CC improves by 4.8% and 28.2% compared with CE at $\epsilon = 0.2$ and $\epsilon = 0.4$, respectively, on ISIC. The reason is that the cross-entropy loss easily results in the overfitting issue on corrupted labels.

3.3 Ablation Study

To verify the effectiveness of the noise cleaner, we compare our method with the following comparison methods: 1) W/O NC: without noise cleaner, and 2) MLP: replace GCN with Multi-Layer Perceptron, *i.e.,* without considering the relationship among samples. Due to the space limitation, we only show results on ISIC, which is listed in the first and second rows of Table 2. The methods with noise cleaner (*i.e.,* MLP and CNLC) outperform the method without noise cleaner W/O NC. For example, CNLC improves by 4.2% compared with W/O NC at $\epsilon = 0.4$. Thus, the noise cleaner plays an critical role in CNLC. Additionally, CNLC obtains better performance than MLP because it considers the relationship among samples. Both of the above observations verify the conclusion mentioned in the last section again.

To verify the effectiveness of the resistance loss in Eq. (1), we remove the second term in Eq. (1) to have a new comparison method CNLC-RL and list the results in the third row of Table 2. Obviously, CNLC outperforms CNLC-RL. For example, CNLC improves by 1.0% and 2.3%, respectively, compared to CNLC-RL, in terms of four evaluation metrics at $\epsilon = 0.2$ and $\epsilon = 0.4$. The reason is that the robustness loss can prevent the model from overfitting on corrupted labels, and thus boosting the model robustness. This verifies the effectiveness of the resistance loss defined in Eq. (1) for medical image analysis, which has been theoretically and experimentally verified in the application of natural images [12].

4 Conclusion

In this paper, we proposed a novel co-assistant framework, to solve the problem of DNNs with corrupted labels for medical image analysis. Experiments on three medical image datasets demonstrate the effectiveness of the proposed framework. Although our method has achieved promising performance, its accuracy might be further boosted by using more powerful feature extractors, like pre-train models on large-scale public datasets or some self-supervised methods, *e.g.,* contrastive learning. In the future, we will integrate these feature extractors into the proposed framework to further improve its effectiveness.

Acknowledgements. This paper is supported by NSFC 62276052, Medico-Engineering Cooperation Funds from University of Electronic Science and Technology of China (No. ZYGX2022YGRH009 and No. ZYGX2022YGRH014).

References

1. Arpit, D., et al.: A closer look at memorization in deep networks. In: International Conference on Machine Learning, pp. 233–242 (2017)
2. Chen, Y., Shen, X., Hu, S.X., Suykens, J.A.: Boosting co-teaching with compression regularization for label noise. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2688–2692 (2021)
3. Codella, N.C., et al.: Skin lesion analysis toward melanoma detection: a challenge at the 2017 international symposium on biomedical imaging (ISBI), hosted by the international skin imaging collaboration (ISIC). In: International Symposium on Biomedical Imaging, pp. 168–172 (2018)
4. Franceschi, L., Frasconi, P., Salzo, S., Grazzi, R., Pontil, M.: Bilevel programming for hyperparameter optimization and meta-learning. In: International Conference on Machine Learning, pp. 1568–1577 (2018)
5. Guo, K., Cao, R., Kui, X., Ma, J., Kang, J., Chi, T.: LCC: towards efficient label completion and correction for supervised medical image learning in smart diagnosis. J. Netw. Comput. Appl. **133**, 51–59 (2019)
6. Han, B., et al.: Co-teaching: robust training of deep neural networks with extremely noisy labels. In: Advances in Neural Information Processing Systems (2018)
7. Jiang, L., Zhou, Z., Leung, T., Li, L.J., Fei-Fei, L.: Mentornet: learning data-driven curriculum for very deep neural networks on corrupted labels. In: International Conference on Machine Learning, pp. 2304–2313 (2018)
8. Laine, S., Aila, T.: Temporal ensembling for semi-supervised learning. arXiv preprint arXiv:1610.02242 (2016)
9. Liu, J., Li, R., Sun, C.: Co-correcting: noise-tolerant medical image classification via mutual label correction. IEEE Trans. Med. Imaging **40**(12), 3580–3592 (2021)
10. Liu, R., Gao, J., Zhang, J., Meng, D., Lin, Z.: Investigating bi-level optimization for learning and vision from a unified perspective: a survey and beyond. IEEE Trans. Pattern Anal. Mach. Intell. **44**(12), 10045–10067 (2021)
11. Lu, Y., He, W.: Selc: self-ensemble label correction improves learning with noisy labels. arXiv preprint arXiv:2205.01156 (2022)
12. Shi, X., Guo, Z., Li, K., Liang, Y., Zhu, X.: Self-paced resistance learning against overfitting on noisy labels. Pattern Recogn. **134**, 109080 (2023)

13. Spanhol, F.A., Oliveira, L.S., Petitjean, C., Heutte, L.: A dataset for breast cancer histopathological image classification. IEEE Trans. Biomed. Eng. **63**(7), 1455–1462 (2015)
14. Tanaka, D., Ikami, D., Yamasaki, T., Aizawa, K.: Joint optimization framework for learning with noisy labels. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 5552–5560 (2018)
15. Valanarasu, J.M.J., Patel, V.M.: Unext: MLP-based rapid medical image segmentation network. In: Wang, L., Dou, Q., Fletcher, P.T., Speidel, S., Li, S. (eds.) MICCAI 2022. LNCS, vol. 13435, pp. 23–33. Springer, Cham. (2022). https://doi.org/10.1007/978-3-031-16443-9_3
16. Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., Summers, R.M.: Chestx-ray8: hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 2097–2106 (2017)
17. Wei, H., Feng, L., Chen, X., An, B.: Combating noisy labels by agreement: a joint training method with co-regularization. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 13726–13735 (2020)
18. Xiao, T., Zeng, L., Shi, X., Zhu, X., Wu, G.: Dual-graph learning convolutional networks for interpretable Alzheimer's disease diagnosis. In: Wang, L., Dou, Q., Fletcher, P.T., Speidel, S., Li, S. (eds.) MICCAI 2022. LNCS, vol. 13438, pp. 406–415. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-16452-1_39
19. Yu, S., et al.: Multi-scale enhanced graph convolutional network for early mild cognitive impairment detection. In: Martel, A.L., et al. (eds.) MICCAI 2020. LNCS, vol. 12267, pp. 228–237. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-59728-3_23
20. Yu, X., Han, B., Yao, J., Niu, G., Tsang, I., Sugiyama, M.: How does disagreement help generalization against label corruption? In: International Conference on Machine Learning, pp. 7164–7173 (2019)
21. Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O.: Understanding deep learning (still) requires rethinking generalization. Commun. ACM **64**(3), 107–115 (2021)
22. Zheng, G., Awadallah, A.H., Dumais, S.: Meta label correction for noisy label learning. In: AAAI Conference on Artificial Intelligence, vol. 35, pp. 11053–11061 (2021)