2017 IEEE 12th International Conference on Automatic Face & Gesture Recognition

Self-Error-Correcting Convolutional Neural Network for Learning with Noisy Labels

Xin Liu[1,2], Shaoxin Li[3], Meina Kan[1,2], Shiguang Shan[1,2,4], Xilin Chen[1,2]
[1] Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS),
Institute of Computing Technology, CAS, Beijing, 100190, China
[2] University of Chinese Academy of Sciences, Beijing 100049, China
[3] Tencent BestImage Team, Shanghai, 100080, China
[4] CAS Center for Excellence in Brain Science and Intelligence Technology

Abstract— **Convolutional Neural Network (CNN) together with large-scale labeled data has achieved the state-of-the-art accuracy in various computer vision tasks. In real-world settings, however, the labels of large scale data can be noisy, which shall seriously degenerate the performance of CNN. In this work, we propose a self-error-correcting CNN (SEC-CNN) to deal with the noisy labels problem, by simultaneously correcting the improbable labels and optimizing the deep model. Specifically, the SEC-CNN provides an opportunity to correct a wrong label by developing a confidence policy to switch between the label of the sample and the max-activated output neuron of the CNN. Based on the assumption that the deep model is more and more accurate during the training, the confidence policy relies more on the given labels at the beginning stages, but tends to believe that the max-activated neuron of the learned network is reliable. SEC-CNN enables CNN learning to be effective even with 80% noisy labels. Extensive experimental results on MNIST, CIFAR-10, ImageNet and CCFD face dataset demonstrate the effectiveness of the proposed method in dealing with noisy labels.**

I. Introduction

Convolutional Neural Network (CNN) with large-scale labeled dataset has greatly promoted various computer vision tasks, such as object recognition [1], face recognition [2], attributes learning [3] and scene classification [4]. The success of CNN relies on large-scale labeled dataset, e.g. ImageNet [5], WebFace [6], Places [4]. In many real-world applications, however, big data with accurate labels is pretty hard or even impossible to obtain, as human labeling is time consuming or needs experts and special devices [7].

Fortunately, in many real-world scenarios, it is easy to obtain large-scale data via some semi-automatic tools, such as web search engines using keywords. However, labels obtained in this way may not be completely reliable. For example, the labels may be flipped, i.e. a sample belong to one class is mistakenly labeled as another [8]. In this circumstance, all the samples are labeled but we do not know whether any label is correct or not. Existing CNN learning methods will degenerate or even fail to work due to wrong label in computing the loss function. Therefore, special efforts are needed to deal with these scenarios. In this work, we propose a self-error-correcting CNN (SEC-CNN) to model noisy label in a unified end-to-end learning framework, as illustrated in Fig. 1.

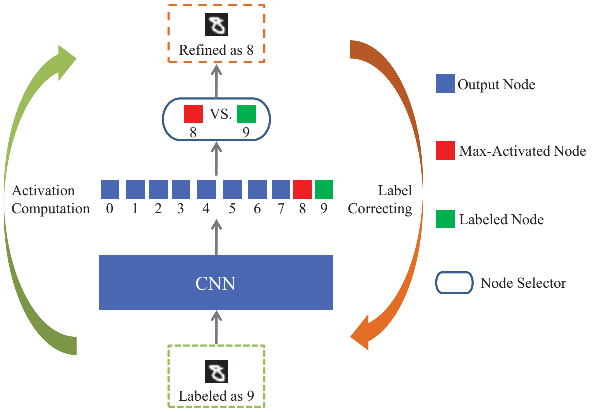

Fig. 1. The basic idea of the proposed self-error-correcting CNN (SEC-CNN) training framework for learning with noisy labels.

Overall speaking, to deal with noisy labels, SEC-CNN jointly refines the noisy labels and updates the CNN network. The basic assumption of our method is that a reasonable CNN can be achieved before completely fitting to the noisy labels. By following a gradually refining confidence policy, we firstly set a higher probability to rely more on the noisy labels, and then decrease the probability to rely less on the noisy labels while put more confidence on the output labels of CNN, i.e. correcting the noisy labels. As the labels become more and more accurate, the CNN also performs better and better.

The contributions of this work can be summarized as:

1) We propose a self-error-correcting CNN (SEC-CNN) training framework to deal with noisy labels, which jointly refines noisy labels and learns deep model following a gradually refining confidence policy, resulting in both corrected labels and better CNN model.

2) The proposed method achieves very promising results on MNIST [9], CIFAR-10 [10], ImageNet [5] and large-scale real-world face dataset CCFD, with favorable robustness to large proportion of noisy labels even up to 80%.

The rest of this paper is organized as follows. Section 2 reviews the related works on deep learning with noisy labels. Section 3 details the proposed SEC-CNN method and section 4 presents the experimental results. Section 5 concludes this work.

978-1-5090-4023-0/17 $31.00 © 2017 IEEE
DOI 10.1109/FG.2017.22

II. Related Works

There are many literatures on learning with noisy labels. One can refer to the survey of learning in the presence of noisy label in [8] for more background. In this section, we briefly review deep learning with noisy labels, and highlights the novelty of our SEC-CNN.

Classical approaches on learning with noisy labels can be roughly divided into three categories: 1) noise-robust models, 2) data cleansing methods and 3) noise-tolerant learning algorithms [8]. For the noise-robust methods, noisy labels are feed into models directly in process of learning. Theoretical analysis showed that most supervised loss functions are not completely robust to label noise [18] and noise-robust models relied on avoiding over-fitting [17] to handle noise. For data cleansing algorithms, the basic idea is to remove samples that appear to have incorrect labels [19], [20]. The shortcoming of these approaches is that it is difficult to distinguish informative hard samples from harmful mislabeled ones [21]. For noise-tolerant learning algorithms, modeling the noise distribution [15], [16] or developing noise-robust loss functions [22], [23] are two popular schemes.

Considering handling label noise in deep network, Reed *et al.* [12] proposed a generic way to handle noise by augmenting the objective function with a notion of label consistency. Sukhbaatar *et al.* [13] introduced an extra noise layer into the network which adapted the network outputs to match the noisy label distribution. Xiao *et al.* [11] proposed to model the relationships between images, class labels and label noises in a probabilistic graphical model and further integrated it into an end-to-end deep learning system. Fu *et al.* [14] proposed to embed the feature map of the last deep layer into a new affinity representation and minimized the discrepancy between the affinity representation and its low-rank approximation to softly limit the contribution of noisy labels.

Sharing a similar motivation of modeling the distribution of label noise, our work gradually estimates the labels of training samples and selects the noisy label or max-activated label to calculate the loss in each training mini-batch according to a gradually refining Bernoulli distribution. Different from Sukhbaatar *et al.*'s work [13], our method does not model the label-level noise distribution and focuses on refining the label of each training sample coupled with the CNN, leading to a more effective scheme. Different from Reed *et al.*'s [12] work which always relies on the noisy label and uses the network output as a regularization, our method gradually decreases the confidence of noisy labels and put more confidence on the deep model output, which can gradually refine the labels and also the CNN. Different from Xiao *et al.*'s work [11] which supposes part training samples have accurate labels, all the training samples in this work are noisy labeled, which is more general. Different form Fu *et al.*'s work [14], our approach would correct the noisy labels rather than merely limit their contributions to the loss function.

III. Proposed Method

In this part, we present the proposed method for learning CNN with noisy labels.

A. Self-Error-Correcting Convolutional Neural Network

In the self-error-correcting CNN training framework, the standard softmax loss is modified to adaptively switch between the noisy label or max-activated neuron according to a simple but effective confidence policy parameterized by a gradually refining Bernoulli distribution.

Self-Error-Correcting Softmax Loss: It should be noticed that for the standard softmax loss, the ground-truth label is supposed to be known and accurate. However, in the noisy label learning scenario, the given label l_i may not be accurate. In this work, we formulate a self-error-correcting softmax loss as follows:

$$L = -\frac{1}{N}\sum\nolimits_{i=1}^{N}\log(\hat{p}_{i\hat{l}_i}), \tag{1}$$

where $\hat{l}_i$ denotes the estimated label of x_i, N is the mini-batch size, and $\hat{p}_{ij}$ is calculated by softmax function of the C-dimension output of network and denotes the probability of the i-th sample belongs to class j.

$$\hat{p}_{ij} = \frac{\exp(o_{ij})}{\sum_{j=0}^{C-1}\exp(o_{ij})}. \tag{2}$$

Note that, the estimated label $\hat{l}_i$ can be switched between the given label l_i of the sample and the max-activated output m_i of the CNN:

$$m_i = \operatorname*{argmax}_{j} \hat{p}_{ij}, \ \ j = 0, 1, ..., C-1. \tag{3}$$

Confidence Policy: In this work, the switching behavior is controlled by a well-designed confidence policy. The basic assumption is that the deep model will become more and more accurate in the training procedure. At the beginning of deep network training, it is not a practical idea to set $\hat{l}_i = m_i$, as the deep classifier layer is initialized randomly. In this work, we introduce C_t as the confidence for the noisy label l_i in the t-th training iteration. C_t is a crucial issue in the self-error-correcting learning as it determines whether to rely more on the noisy label or on the output label of the deep model. However, since the label is noisy, it is not a good idea to keep C_t unchanged either. Otherwise, the CNN may overfit to the wrong label, which will degenerate the final performance of the CNN. So, after the CNN fits better on the training data as the iteration of optimization increases, the confidence on the network output m_i should be increased accordingly. As the CNN learning is coupled with the label correcting, we develop a polynomial confidence policy to elaborately cooperate with the CNN in each iteration:

$$C_t = C_0 * (1 - t/T)^{\lambda}, \tag{4}$$

where C_0 denotes the initial confidence, t denotes the current iteration of training, T denotes the total number of iterations

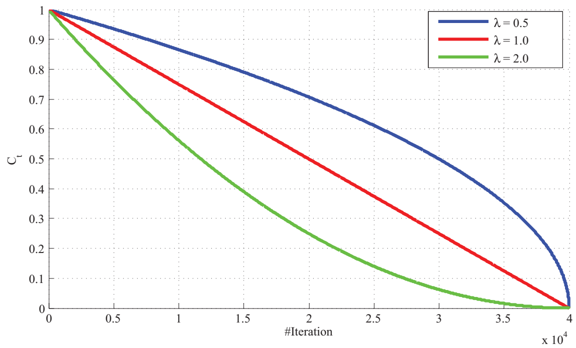

Fig. 2. Examples of confidence policy curve with different exponent values λ.

and λ denotes the exponent. The basic principal under the confidence policy design is that with the increase of training iteration, we will put more confidence on the deep model output and less rely on the noisy labels. In Fig. 2, the confidence policy curves with different exponent values are presented. We will explore the the parameter setting of exponent values in the experimental sections.

The estimated label $\hat{l}_i$ can be estimated following a Bernoulli distribution:

$$\hat{l}_i = \begin{cases} l_i & \text{if } x = 1,\ x \sim B(1; C_t). \\ m_i & \text{if } x = 0,\ x \sim B(1; C_t). \end{cases} \tag{5}$$

In general, $\hat{l}_i$ is randomly determined by the Bernoulli distribution $B(1; C_t)$ with increasing probability to select the max-activated label.

The backward operation of the self-error-correcting softmax loss will be:

$$\nabla_{ij} = \begin{cases} \hat{p}_{ij} - 1 & \text{if } j = \hat{l}_i, \\ \hat{p}_{ij} & \text{otherwise}, \end{cases} \tag{6}$$

where ∇_{ij} denotes the gradient value of the j-th output node of x_i in the current mini-batch.

Training Pipeline: The overall training pipeline of learning CNN with noisy labels is also presented in Algorithm 1. The proposed approach gradually refine the noisy labels and further use them to update the CNN in each training iteration, formulating an end-to-end learning framework.

B. Discussions of SEC-CNN

How SEC-CNN works: In SEC-CNN, the optimization of CNN and the refinement of the noisy label are coupled in each iteration, thus they can benefit each other more than some other models such as [15], [16], owing to the gradual gradient descent optimization of CNN. Take [16] for example, it also iteratively updates the classification model and the refinement of noisy labels. However in each iteration, given a previous refinement the classification model is achieved analytically which perfectly fits the current refinement, so the classification model will be misled if the refinement is inaccurate, leading to further poor refinement. On the contrary, in each iteration of our scheme, given the previous refinement the classification model (i.e. the CNN) is updated forward in a very small step following the gradient descent scheme, so the updated CNN model considers the refinement noisy label, but only slightly deviates from the previous model, which means the CNN model will not be misled even if the current refinement is inaccurate. As a result, the proposed SEC-CNN steadily and gradually incorporate those label refinement and can induce better label refinement and CNN model.

Algorithm 1 Self-error-correcting CNN (SEC-CNN) Training Framework.

```
Input: CNN Network, mini-batch $N$ and total number of iteration $T$.
Output: Learned CNN Network.
Set $t = 0$ and initialize CNN$_t$ randomly.
while $t < T$ do
    // Forward pass.
    Calculate the network output using CNN$_t$.
    // Noisy label refinement.
    Calculate $C_t$ according to Eqn. (4).
    for each $x_i$ in $N$ do
        Estimate $\hat{l}_i$ according to Eqn. (5).
    end for
    //Backward pass.
    for each $x_i$ in $N$ do
        Calculate $\nabla_{ij}$ according to Eqn. (6).
    end for
    Set $t = t + 1$.
    Update CNN$_t$ using back propagation.
end while
return Learned CNN$_T$.
```

Role of Normalization Layer: One of the key issues of the SEC-CNN is the rationality of the max-activated output m_i of the CNN. In a plain CNN, using the max-activated output as estimated label is risky. In fact, when training a plain CNN from scratch, we observe severe "label dominance" phenomenon among the output neurons of CNN. As shown in Fig. 3(b), the max activation has the risk of appearing in a few neurons or even single neuron. Thus, these neurons will soon be over-enhanced and result in a degenerated CNN.

To solve this problem, we add a normalization layer (NL) before the softmax loss layer. NL is similar to batch normalization layer [25], but without scaling and shifting of the normalized value. Specifically, the normalization layer aims at normalizing the output of each network node to have zero mean and unit variance, which directly ensures the "label dominance" phenomenon not happen, as shown in Fig. 3(a).

Although the BN layer is already proven to be very effective [25], [26], to the best of our knowledge, none of the previous works uses it to normalize the output neurons. Meanwhile, besides preventing "label dominance", we found adding the normalization layer after all the convolutional and inner-product layers also speeds up the convergence, resulting a more accurate initial model for label estimation. This is the same to the original idea of conducting normalization in the neural networks [24], [25]. In our implementation of SEC-CNN, NL layers are added after all the convolutional and inner-product layers and before the final softmax loss layer.

IV. Experiments

In this section, we investigate our approach on four dataset, i.e., three well-known image classification datasets including MNIST, CIFAR-10 and ImageNet, and one large-scale real-world face dataset, i.e., Chinese Celebrity Face Dataset (CCFD). On MNIST, CIFAR-10 and ImageNet, we conduct experiments to examine the robustness of the proposed method in the presence of different proportions of noisy labels. On CCFD, we conduct experiments in the presence of noisy labels.

A. Dataset

MNIST: MNIST handwritten digits dataset is a well-known image classification benchmark. It has 10 classes with 60,000 training images and 10,000 test images.

CIFAR-10: The CIFAR-10 dataset consists of 60,000 samples in 10 object classes, with 6,000 images per class. It has 50,000 training images and 10,000 test images.

ImageNet: The ImageNet dataset consists of more than 1 million train samples and 50,000 validation samples in 1,000 object classes. Now it is the state-of-the-art benchmark for object classification. In this paper, the ImageNet2012 dataset is adopted [5].

CCFD: The Chinese Celebrity Face Database (CCFD) is a large-scale real-world face dataset collected by ourselves. This dataset consists of 1,001 subjects of 263,696 images, with two subsets for training and testing. The training set contains 701 subjects of 171,792 images and the test set contains 300 subjects of 91,904 images. Facial images in CCFD are collected in real-world environments from the internet and has large variations in age, expression, light, occlusion and pose. Fig. 4 shows a few exemplar subjects in CCFD. All the samples are normalized to 256×256 colorful images in advance. Unlike the classical image classification settings, the training set and the test set have no overlap in class labels. We take this dataset to evaluate the generalizabiltiy of the learned feature of CNN with noisy labels labels.

B. Experiments on MNIST

In this section, we conduct experiments on MNIST dataset. To simulate the noisy label scenarios, we randomly flip the label of a training sample to other 9 classes w.r.t. a given probability, which is also the proportion of the noisy labels.

Parameters Settings: We take the standard LeNet [28] network for the experiments on MNIST. SGD is utilized to optimize the LeNet. In this experiments, we set the base_lr as 0.01, mini-batch size as 128, total iteration as $10,000$. The learning rate is decreased by the polynomial policy with power value equals to 0.5. For the proposed self-error-correcting learning framework, we set the initial noisy label confidence C_0 as 1.0 and the exponent value as 0.5 or 1.0.

Fig. 4. Example subjects in Chinese Celebrity Face Dataset (CCFD). Facial images in one row represent one distinct subject.

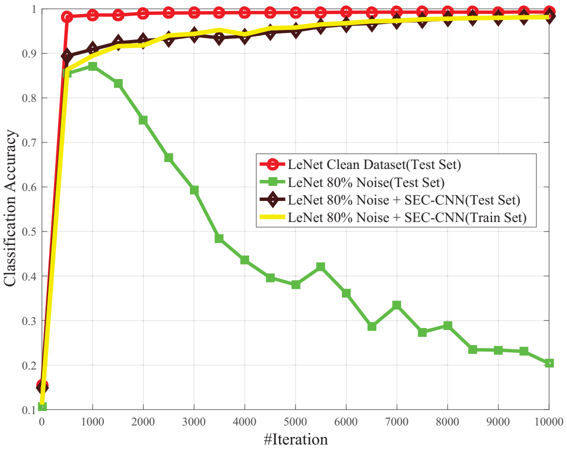

Fig. 5. Analysis of SEC-CNN on MNIST with 80% label noise.

Results on Different Proportions of Label Noise: As shown in Table I, without considering the noise-tolerance, the performance of all the two baseline networks degrades seriously. On the contrary, benefiting from the self-error-correcting learning framework our proposed method significantly improves the robustness of CNN to label noise even up to 80% proportion, resulting in less than 1% performance degradation than the corresponding network learned without any noise.

Analysis of how SEC-CNN Works: Our basic assumption is that the CNN can learn a better model than random guess at the early stage of training even with large proportion of label noise. As shown in Fig. 5, even with 80% label noise, CNN without noisy label learning obtains reasonable accuracy at the first 20% iterations which supports our assumption experimentally. It also indicates that CNN network can learn reasonable models with noisy ground-truth labels at the beginning of training, which is the basis of the success of our method. Meanwhile, SEC-CNN also corrects the corrupted label and 98.13% labels are finally corrected even under the 80% label noise.

Extensions of SEC-CNN: We try to extend SEC-CNN to hold random noise by setting auxiliary noise node in the last fully connected layer. In MNIST, we randomly add digit 9

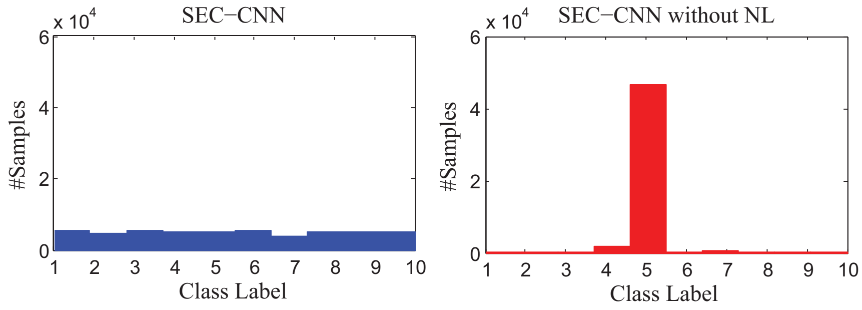

Fig. 3. Visualization of the max-activated label distributions of training samples in CIFAR-10 with CIFARQuick network following [14].

TABLE I

CLASSIFICATION ACCURACY ON MNIST WITH DIFFERENT PROPORTIONS OF LABEL NOISE. THE BEST RESULT IN EACH COLUMN IS SHOWN IN BOLD.

Network Architecture	Proportion of Noisy Label								
	clean	10%	20%	30%	40%	50%	60%	70%	80%
LeNet	99.03	95.18	91.07	84.85	77.57	69.32	51.26	48.80	24.57
LeNet+SEC-CNN$_{\lambda=1}$	99.26	**99.07**	**99.17**	98.98	**98.86**	**98.83**	98.63	**98.67**	**98.41**
LeNet+SEC-CNN$_{\lambda=0.5}$	**99.27**	99.13	99.01	98.89	98.76	98.53	98.44	97.96	97.74

as unknown classes to the rest digit 0 to 8 classes in the database as random noise, and wet set one noise node to handle this kind of label noise. Considering the ten classes classification of digit 0 to 9 on MNIST, the plain LeNet achieves 88.97% mean accuracy and SEC-CNN achieves 98.70% mean accuracy. This experiment demonstrates that SEC-CNN has the potential to be extended to handle unknown classes mislabeled as known classes.

We also add an experiment on MNIST by random setting the noise ratio of a class between 10% and 90%, and experimental results shows SEC-CNN also achieves 98.31% mean accuracy under this non-uniform noise scenario.

C. *Experiments on CIFAR-10*

In this section, we conduct experiments on CIFAR-10. Consistent with the MNIST settings, the label noise proportion is equal to the probability of flipping the label of one training sample to other 9 classes randomly.

Parameters Settings: On CIFAR-10, we take the 5-layer CIFARQuick network following [14] for the experiments. We adopt the same SGD configuration and same SEC-CNN configuration as that used in the MNIST experiments except the total iteration as 20,000.

Comparisons with State-of-the-art Methods: Table II presents the classification accuracy on CIFAR-10 test set with different proportions of label noise. The NL with SEC-CNN outperforms the state-of-the-art methods when the label noise is no less than 20%. When the label noise is 0% and 10%, our method performs a little worse, and it is caused by a side effect of the self-error-correcting learning framework as it has the risk of taking hard samples as label noise.

D. *Experiments on ImageNet*

In this section, we conduct experiments on ImageNet. In this experiment, the first 100 classes are adopted and the label noise proportion is equal to the probability of flipping the label of one training sample to other 99 classes randomly. Considering the complexity of ImageNet, CaffeNet network [27] is employed.

TABLE III

TOP-1 CLASSIFICATION ACCURACY ON THE FIRST 100 CLASSES OF IMAGENET2012 WITH DIFFERENT PROPORTIONS OF LABEL NOISE. THE BEST RESULT IN EACH COLUMN IS SHOWN IN BOLD.

Network Architecture	Proportion of Noisy Label			
	clean	20%	50%	80%
CaffeNet	**61.42**	53.00	0.01	0.01
CaffeNet+SEC-CNN$_{\lambda=1}$	61.40	55.30	43.98	8.22
CaffeNet+SEC-CNN$_{\lambda=0.5}$	61.28	**55.40**	**47.30**	**24.70**

Parameters Settings: In this experiment, we adopt the same SGD configuration and same SEC-CNN configuration as those used in the MNIST experiments. We set the total iteration as 20,000.

Results under Different Proportions of Label Noise: Table IV presents the top-1 classification accuracy on the 5,000 validation samples on the first 100 classes of ImageNet with different proportions of label noise. Experimental results demonstrate that SEC-CNN can handle large proportion of label noise even up to 80%.

Analysis of Parameters in Confidence Policy: The exponent value λ of the confidence policy controls the rate of decline of the confidence value C_t. Overall, results in Table I, Table II and Table III demonstrate that performance of SEC-CNN is somewhat related to λ and choosing λ via validation set will be a practical solution.

E. *Experiments on Large-scale Face Dataset*

In this section, we further evaluate the proposed method on CCFD in the presence of noisy labels. On this dataset, the 8-

TABLE II

Classification accuracy on CIFAR-10 with different proportions of label noise. The best result in each column is shown in bold.

Network Architecture	Proportion of Noisy Label								
	clean	10%	20%	30%	40%	50%	60%	70%	80%
Plain CNN [14]	81.24	77.79	71.97	65.09	55.65	45.60	36.65	25.02	19.46
Bottom-Up [13]	81.16	78.28	73.36	68.26	61.63	55.83	47.33	37.12	30.81
Method of [11]	–	–	–	69.81	66.76	63.00	–	–	–
NRCNN [14]	**81.60**	**79.39**	76.21	72.81	68.79	63.01	54.78	45.48	35.43
CIFARQuick	80.84	78.01	73.25	69.91	61.65	55.28	46.34	34.59	23.97
CIFARQuick+SEC-CNN$_{\lambda=1}$	78.48	78.13	74.98	73.03	72.72	68.04	63.30	52.01	35.79
CIFARQuick+SEC-CNN$_{\lambda=0.5}$	79.95	79.02	**76.91**	**74.98**	**73.02**	**70.67**	**65.30**	**57.57**	**43.86**

TABLE V

Verification Rate @FAR = 0.1% on CCFD with different confidence policy exponent value at 50% label noise.

Network Architecture	Exponent Value (λ)		
	0.5	1.0	2.0
CaffeNetNL+SEC-CNN	24.25	**25.43**	2.86

layer CaffeNet network [27] is employed for all experiments considering the large scale of training data.

Parameters Settings: SGD is utilized to optimize the CaffeNet. In this experiments, we set the base_lr as 0.01, mini-batch size as 192, and total iteration as $20,000$. The learning rate is decreased according to the polynomial policy with gamma value equals to 0.5. For the verification test, we extract the features of FC7 layer and adopt cosine similarity.

Results under Different Proportions of Label Noise: Consistent with the MNIST and CIFAR-10 experimental settings, we use a random label permutation and the label noise proportion is equal to the probability of flipping the label of one training sample to other 700 classes in the training set randomly. Table IV presents the verification rate on CCFD with different proportions of label noise. As seen from the comparisons, the same conclusions can be obtained:

The self-error-correcting learning framework significantly improves the robustness of the CNN network to noise in the real-world face dataset, especially when with a high proportion of noise. In the presence of more than 40% proportion of label noise, the CaffeNet fails to convergence while SEC-CNN still works robust toward label noise. While when there is 80% noise, SEC-CNN also fails to convergence due to the poor initialization of the deep model with large proportion of label noise. As in this circumstance, the assumption that CNN can learn a reasonable of model when label is noisy isnot established anymore.

Analysis of Parameters in Confidence Policy: Table V compares the confidence policy with different exponent values. When the exponent value equals to 2, the confidence decreases too quickly at the beginning of training, resulting in a bad local optimal label refinement and poor performance. And it is recommended to determine λ via the cross-validation.

V. Conclusions

In this work, we propose the self-error-correcting CNN (SEC-CNN) learning framework to deal with the problem of noisy labels. We evaluate the proposed approaches in both image classification datasets and large-scale real-world face dataset, with respect to different proportions of noisy labels. We also explore the parameters setting of confidence policy in SEC-CNN. Experimental evaluations on MNIST, CIFAR-10, ImageNet and CCFD face dataset demonstrate that our model is robust to label noise even up to 80% proportion. For the future work, we will extend SEC-CNN to handle missing label scenario.

VI. Acknowledgement

This work was partially supported by 973 Program under contract No. 2015CB351802, Natural Science Foundation of China under contracts Nos. 61390511, 61650202, 61402443, 61272321, and the Strategic Priority Research Program of the CAS (Grant XDB02070004).

References

[1] Krizhevsky, A., Sutskever, I., Hinton, G.E, “Imagenet classification with deep convolutional neural networks”, *Advances in Neural Information Processing Systems*, pp 1097-1105, 2012

[2] Taigman, Y., Yang, M., Ranzato, M., Wolf, L, “Deepface: Closing the gap to human-level performance in face verification”, *IEEE Conference on Computer Vision and Pattern Recognition*, pp 1701-1708, 2014

[3] Zhang, N., Paluri, M., Ranzato, M., Darrell, T., Bourdev, L, “Panda: Pose aligned networks for deep attribute modeling”, *IEEE Conference on Computer Vision and Pattern Recognition*, pp 1637-164, 2014

[4] Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A, “Learning deep features for scene recognition using places database”, *Advances in Neural Information Processing Systems*, pp 487-495, 2014

[5] Russakovsky, Olga and Deng, Jia and Su, Hao and Krause, Jonathan and Satheesh, Sanjeev and Ma, Sean and Huang, Zhiheng and Karpathy, Andrej and Khosla, Aditya and Bernstein, Michael and others, “Imagenet large scale visual recognition challenge”, *International Journal of Computer Vision*, vol. 115, pp 211-252, 2015

[6] Yi, D., Lei, Z., Liao, S., Li, S.Z, “Learning face representation from scratch”, *arXiv preprint arXiv:1411.7923*, 2014

[7] Zhu, X, “Semi-supervised learning literature survey”, *Technical Report 1530, Department of Computer Sciences, University of Wisconsin, Madison*, 2005

[8] Frénay, Benoît and Verleysen, Michel, “Classification in the presence of label noise: a survey”, *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, pp 845-869, 2014

[9] LeCun, Yann and Bottou, Léon and Bengio, Yoshua and Haffner, Patrick, “Gradient-based learning applied to document recognition”, *Proceedings of the IEEE*, vol. 86, pp 2278-2324, 1998

TABLE IV

VERIFICATION RATE @FAR = 0.1% ON CCFD WITH DIFFERENT PROPORTIONS OF LABEL NOISE.

Network Architecture	Proportion of Noisy Label								
	clean	10%	20%	30%	40%	50%	60%	70%	80%
CaffeNet	33.45	19.84	17.06	14.91	13.20	Fails	Fails	Fails	Fails
CaffeNet+SEC-CNN$_{\lambda=1}$	32.89	**32.42**	**31.50**	**30.80**	**32.44**	**25.43**	**28.01**	4.10	Fails
CaffeNet+SEC-CNN$_{\lambda=0.5}$	**33.15**	30.75	30.79	28.80	27.80	24.25	22.88	**19.27**	Fails

[10] Krizhevsky, Alex and Hinton, Geoffrey, "Learning multiple layers of features from tiny images", *Citeseer*, 2009

[11] Xiao, Tong and Xia, Tian and Yang, Yi and Huang, Chang and Wang, Xiaogang, "Learning from Massive Noisy Labeled Data for Image Classification", *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp 2691-2699

[12] Reed, Scott and Lee, Honglak and Anguelov, Dragomir and Szegedy, Christian and Erhan, Dumitru and Rabinovich, Andrew, "Training Deep Neural Networks on Noisy Labels with Bootstrapping", *arXiv preprint arXiv:1412.6596*, 2014

[13] Sainbayar Sukhbaatar and Rob Fergus, "Learning from Noisy Labels with Deep Neural Networks", *CoRR*, 2014

[14] Xiao, Tong and Xia, Tian and Yang, Yi and Huang, Chang and Wang, Xiaogang, "Learning from Massive Noisy Labeled Data for Image Classification", *IEEE Conference on Computer Vision*, pp 2691-2699, 2015

[15] Lawrence, Neil D and Schölkopf, Bernhard, "Estimating a kernel Fisher discriminant in the presence of label noise", *International Conference on Machine Learning*, pp 306-313, 2001

[16] Wang, Dong and Tan, Xiaoyang, "Robust distance metric learning in the presence of label noise", *AAAI Conference on Artificial Intelligence*, pp 1321-1327, 2014

[17] Teng, Choh-Man, "A Comparison of Noise Handling Techniques", *FLAIRS Conference*, pp 269-273, 2001

[18] Bartlett, Peter L and Jordan, Michael I and McAuliffe, Jon D, "Convexity, classification, and risk bounds", *Journal of the American Statistical Association*, vol. 101, pp 138-156, 2006

[19] Barandela, Ricardo and Gasca, Eduardo, "Decontamination of training samples for supervised pattern recognition methods", *Advances in Pattern Recognition*, pp 621-630, 2000

[20] Miranda, André LB and Garcia, Luís Paulo F and Carvalho, André CPLF and Lorena, Ana C, "Use of classification algorithms in noise detection and elimination", *Hybrid Artificial Intelligence Systems*, pp 417-424, 2009

[21] Guyon, Isabelle and Matic, Nada and Vapnik, Vladimir and others, "Discovering Informative Patterns and Data Cleaning", *AAAI Technical Report WS-94-03*, 1994

[22] Natarajan, Nagarajan and Dhillon, Inderjit S and Ravikumar, Pradeep K and Tewari, Ambuj, "Learning with noisy labels", *Advances in Neural Information Processing Systems*, pp 1196-1204, 2013

[23] Biggio, Battista and Nelson, Blaine and Laskov, Pavel, "Support Vector Machines Under Adversarial Label Noise", *Asian Conference on Machine Learning*, pp 97-112, 2011

[24] LeCun, Yann and Bottou, Léon and Orr, Genevieve B and Müller, Klaus-Robert, "Efficient backprop", *Neural networks: Tricks of the trade*, Springer, pp 9-48, 2012

[25] Ioffe, Sergey and Szegedy, Christian, "Batch normalization: Accelerating deep network training by reducing internal covariate shift", *Internatioal Conference on Machine Learning*, pp 448-456, 2015

[26] He, Kaiming and Zhang, Xiangyu and Ren, Shaoqing and Sun, Jian, "Deep Residual Learning for Image Recognition", *arXiv preprint arXiv:1512.03385*, 2015

[27] Jia, Yangqing and Shelhamer, Evan and Donahue, Jeff and Karayev, Sergey and Long, Jonathan and Girshick, Ross and Guadarrama, Sergio and Darrell, Trevor, "Caffe: Convolutional architecture for fast feature embedding", *Proceedings of the ACM International Conference on Multimedia*, pp 675-678, 2014

[28] LeCun, Yann and Bottou, Léon and Bengio, Yoshua and Haffner, Patrick, "Gradient-based learning applied to document recognition", *Proceedings of the IEEE*, vol.86, pp 2278-2324, 1998