INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Automatic Error Correction for Speaker Embedding Learning with Noisy Labels

Fuchuan Tong[1]*, Yan Liu*[1]*, Song Li*[1]*, Jie Wang*[1]*, Lin Li*[1]*, Qingyang Hong*[2]

[1] School of Electronic Science and Engineering, Xiamen University, China
[2] School of Informatics, Xiamen University, China

lilin@xmu.edu.cn, qyhong@xmu.edu.cn

Abstract

Despite the superior performance deep neural networks have achieved in speaker verification tasks, much of their success benefits from the availability of large-scale and carefully labeled datasets. However, noisy labels often occur during data collection. In this paper, we propose an automatic error correction method for deep speaker embedding learning with noisy labels. Specifically, a label noise correction loss is proposed that leverages a model's generalization capability to correct noisy labels during training. In addition, we improve the vanilla AM-Softmax to estimate a more robust speaker posterior by introducing sub-centers. When applied on the VoxCeleb dataset, the proposed method performs gracefully when noisy labels are introduced. Moreover, when combining with the Bayesian estimation of PLDA with noisy training labels at the back-end, the whole system performs better under conditions in which noisy labels are present.

Index Terms: speaker verification, noisy labels, x-vectors, probabilistic linear discriminant analysis

1. Introduction

Recently, deep neural network (e.g., time delay neural network [1], Res2Net [2]) based embeddings (x-vectors) have become state-of-the-art for speaker verification. Their remarkable success, however, relies on the availability of large-scale and carefully labeled datasets. Unfortunately, noisy labels may occur during data collection. For instance, the NIST SRE18 development set does not include speaker labels and only provides a phone number for each audio sample [3], which inevitably introduces label noise. Although wrong labels can be manually corrected by experts, this is not a practical idea for large data sets since it is time-consuming and costly.

The popular deep embedding speaker verification system often consists of two stages, a neural network embedding front-end and a similarity scoring back-end [4]. The front-end extracts low-dimensional x-vectors, while the back-end computes the similarity between enrollment and test embeddings. For the back-end label noise, Borgström *et al.* [5] propose a novel method for Bayesian estimation of probabilistic linear discriminant analysis (PLDA), which achieves impressive performance when training labels are noisy. Their assumption, however, is based on the presupposition that the front-end network was trained on clean data.

In the presence of noisy labels, the front-end network parameters are updated toward an incorrect gradient descent direction as the incorrect labels exist in the loss function. The loss function leads the network to fit all of the training data and eventually results in poor generalization on clean testing data. Avoiding network overfitting of noisy labels is of particular importance for achieving sufficient generalization capability. While the computer vision literature on label noise is extensive (e.g., face recognition and image classification), this topic has received little attention in speaker verification.

In this paper, we propose a simple yet effective method to handle this issue. The proposed method is derived from the experimental observation that at the beginning stages of training, a model's prediction accuracy on a clean validation set is higher than that on a noisy training set. In other words, a model's prediction accuracy is less affected by noisy labels than its training accuracy. Motivated by this phenomenon, we propose a label noise correction loss that consists of the ground-truth label loss and the predicted label loss with two confidence weights. The weights vary continuously on the basis of the accuracy of network predictions. Since a network becomes more and more accurate during training, the loss accordingly tends to rely more on predicted labels. Furthermore, inspired by the sub-center ArcFace [6], we extend the concept of sub-centers into the AM-Softmax loss to further improve the robustness of label noise. Finally, we investigate the label noise robustness of three kinds of back-ends for speaker verification.

This paper is organized as follows. In Section 2, we present related works from the literature. The proposed approach is explained in Section 3, and experimental results are presented in Section 4. Finally, Section 5 concludes this work.

2. Related works

2.1. Network learning with noisy labels

Types of label noise can be categorized into two categories: closed-set noise and open-set noise. Closed-set noise refers to that noisy samples are mislabeled as other classes that exist in the training set, while in the open-set noise condition, some samples are mislabeled as wrong classes that do not appear in the training set. Since closed-set noise is more challenging [7], in this paper, we only focus on closed-set noise scenarios.

Recently, several studies have been conducted to investigate learning with noisy labels. One can refer to [8] for an overview of recent research on the topic of managing noisy training data. These methods can be roughly divided into four categories: robust loss function, robust architecture, robust regularization, and sample selection. The robust loss function [6, 9, 10, 11] aims to modify its value or design more robust objective functions that avoid overfitting noisy samples; robust architecture [12, 13] aims to model a noise transition matrix from a noisy dataset by using an auxiliary architecture; robust regularization [14, 15, 16] aims to force a network to reduce the overfitting of false labeled samples; and sample selection [17, 18, 19] aims to select true-labeled samples from noisy training data. Concretely, Bekker *et al.* [9] adopted the Expectation Maximization (EM) algorithm to iterate the E-step to estimate the label transi-

http://dx.doi.org/10.21437/Interspeech.2021-2021

tion matrix and the M-step to back-propagate the network. Yao *et al.* [12] proposed the contrastive additive noise network to adjust incorrectly estimated label transition probabilities. The Bootstrap method [20] introduced a consistency objective that effectively relabels data during training. However, most of these methods have high computational cost and cannot be easily integrated into other architectures.

2.2. PLDA estimation with noisy labels

Currently, PLDA [21, 22] is a commonly used back-end scoring approach for speaker recognition. The PLDA models the across-class and within-class variability of speaker representations by using the EM algorithm in a supervised fashion. Labeling errors that occur in the PLDA estimation process would also lead to performance degradation. To overcome this issue, Borgström *et al.* [5] proposed a method for Bayesian estimation of PLDA with noisy labels. For convenience, we refer to this method as NL-PLDA (noise-label PLDA) in the following.

NL-PLDA interprets the ground-truth labels as latent random variables $\mathcal{Z}$ and estimates both the parameters $\{\boldsymbol{\mu}, \mathbf{B}, \mathbf{W}\}$ and the error rate ϵ simultaneously using the EM algorithm in the context of Variational Bayes. Specifically, in the EM iterations, the E-step estimates both the posterior of the true identities and the individual feature distributions simultaneously, and the M-step updates the noise rate, true latent identities and the parameters of NL-PLDA, respectively.

3. Learning with noisy labels

3.1. Label correction loss function

Let us first rethink the deep embedding-based speaker verification systems from a classification perspective. The deep embedding network training process can be formulated as a problem of learning a model $h_\theta(z)$ from a set of batch training samples $\mathcal{D} = \{(z_i, y_i)\}_{i=1}^{B}$, where $y_i \in \{0,1\}^M$ denotes one-hot encoding ground-truth label corresponding to z_i. B is the mini-batch size, and M is the number of total classes. For classification issues with label noise, label y_i would be noisy (i.e. z_i is a noisy sample). Supposing the extracted embedding of z_i is x_i, the parameters of the network are updated by optimizing the following loss function:

$$L = -\frac{1}{B}\sum_{i=1}^{B}\log\left(P_{i,y_i}\right) \tag{1}$$

where P_{i,y_i} denotes the posterior probability of x_i classified to the ground-truth label y_i. In this paper, we term P_{i,y_i} as the *ground-truth label posterior probability*. If adopting the additive margin softmax (AM-Softmax), P_{i,y_i} is formulated as:

$$P_{i,y_i} = \frac{e^{s\left(\cos\left(\theta_{y_i,i}\right)-m\right)}}{e^{s\left(\cos\left(\theta_{y_i,i}\right)-m\right)} + \sum_{j\neq y_i} e^{s\left(\cos\left(\theta_{j,i}\right)\right)}} \tag{2}$$

where $\theta_{j,i}$ is the angle between W_j and x_i. W_j is the j-th class center vector of the fully-connected layer matrix $W \in \mathbb{R}^{M\times d}$. In Section 3.2, we extend the dimensions of W to $(M \times K \times d)$ to compute a more robust P_{i,y_i}.

However, a neural network trained directly with this objective function will overfit to wrong labels, as the ground-truth label might be noisy. Nonetheless, we may observe that a network maintains highly generative performance without memorizing noisy labels at the beginning of the training process; an example of this is shown in Fig. 1, Section 4.2, where the prediction accuracy for clean data is higher than noisy data. This situation indicates that the network is capable of clustering noisy samples into its correct classes; therefore, to leverage this ability, we incorporate a *prediction label posterior probability*, $P_{i,\hat{y}_i}$, into the objective function to prevent fitting into incorrect samples. The subscript $\hat{y}_i \in \{0, 1, ..., M-1\}$ denotes the prediction label for x_i, which is the class j with the max-activated output. Specifically, the loss function in Eq. (1) is extended as

$$L' = -\frac{1}{B}\sum_{i=1}^{B}\left\{(1-\alpha_t)\log\left(P_{i,y_i}\right) + \alpha_t \log\left(P_{i,\hat{y}_i}\right)\right\} \tag{3}$$

where $\alpha_t \in [0,1]$ is the t-th training iteration confidence weight between P_{i,y_i} and $P_{i,\hat{y}_i}$, and it determines whether the loss function relies more on the ground-truth label or the predicted label. This method is similar to Bootstrap [20]. While Bootstrap sets α_t as a fixed small value for all iterations, so it would maintain the effects of noisy labels during training process; thus, the performance is suboptimal since noisy label correction is limited. Conversely, we adopt a dynamic weight for α_t. Since the network parameters are randomly initialized and the predicted labels are less correct, it is not a practical idea to set α_t too large at the beginning of the training stages, and it should not be set too small at the last few training stages. Thus, we formulate α_t as the exponentially increasing function of the iterations, which is written as

$$\alpha_t = \alpha_T \cdot (t/T)^\lambda \tag{4}$$

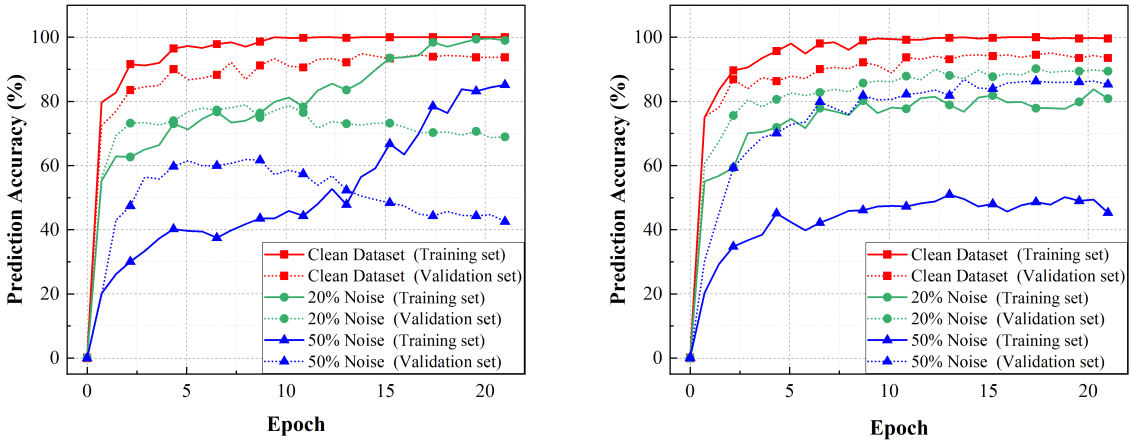
where $\alpha_T \in [0,1]$ represents the confidence weight at the final iteration T, t denotes the number of iterations of the current training, and λ is the exponent that controls the increase rate. With this confidence policy, α_t would dynamically increase from 0.0 to α_T as the number of iterations increases. The basic assumption is that the network will become increasingly accurate in the training evaluation; thus the loss function should accordingly put more reliability on the predicted label. However, at the last few optimization processes, there is a risk that the network may simply predict all samples as belonging to the same class to minimize the loss. To avoid this issue, we add a regularization term [23] into Eq. (3); then, the proposed label noise correction loss L_{total} is formulated as:

$$L_{total} = L' + \beta\frac{1}{M}\sum_{j=0}^{M-1}\log\left(\frac{1}{M\bar{P}_j}\right) \tag{5}$$

where β is the regularization coefficient, and $\bar{P}_j = \frac{1}{B}\sum_{i=1}^{B} P_{i,j}$ is the mean softmax probability for class j. The label regularization term enables the classifier to allocate each sample a probability of being every class, thereby preventing all samples from being assigned to a single class. In this paper, we refer to L_{total} as label noise correction loss.

3.2. Joint label correction and sub-center AM-Softmax

From Eq. (5), one sees that the posterior probability is of particular importance in label noise correction loss. However, the standard AM-Softmax objective function is ill-suited to estimate the posterior probability, as it encourages overfitting error samples. Inspired by sub-center ArcFace [6], we extend the concept of sub-centers to the AM-Softmax loss to improve the robustness of estimation posterior probability on noisy training data and term it as sub-center AM-Softmax. A detailed description of sub-centers is presented in [6]. Briefly, the K sub-centers $W \in \mathbb{R}^{M\times K\times d}$ is introduced for each class. It



(a) (b)

Figure 1: *Comparisons of prediction accuracy between the baseline model (a) and our proposed method (b)*

consists of one dominant sub-class, which contains the majority of clean samples, and the other $(K-1)$ non-dominant sub-classes, which include hard or noisy samples. Reflecting on the formula, the difference between AM-Softmax and sub-center AM-Softmax is that in AM-Softmax, $\cos(\theta_j, i) = W_j^T x_i$, while $\cos(\theta_{j,i}) = \max_k \left(W_{jk}^T, x_i\right), k \in \{1, \cdots, K\}$ in the sub-center AM-Softmax, where $\max_k$ denotes a max-pooling step. By applying the sub-centers, the noisy samples are automatically separated from the majority of clean samples.

4. Experiments

4.1. Datasets and experimental setup

We conducted comprehensive experiments to investigate how the proposed approach performs on speaker verification tasks with noisy labels. We evaluated our approach on the VoxCeleb1 and VoxCeleb2 datasets with different noise scenarios. The VoxCeleb1 and VoxCeleb2 datasets were collected from videos uploaded to YouTube. The VoxCeleb1 development set contains over 100,000 utterances from 1,211 celebrities, while the VoxCeleb2 development set contains over 1 million utterances from 5,994 celebrities, which is much larger than VoxCeleb1. Since VoxCeleb1 is checked manually for any errors and VoxCeleb2 has few labeling errors, they can be regarded as clean datasets ($\epsilon = 0\%$).

In this section, we show our experimental results from three perspectives. First, we conducted ablation studies and showed the prediction accuracy curve during training on the VoxCeleb1 dataset. Second, we performed experiments on the two-fold VoxCeleb1 data augmentation to examine the effectiveness of our method when encountering the hard samples. Third, we further demonstrated the effectiveness on the VoxCeleb2 large dataset.

For all experiments, we randomly split 1,000 recordings (each from a different speaker) from the development set for validation and called it the *validation set*. The remaining was used as the *training set*. Since the validation set was clean, it was set aside to monitor the network training performance. Note that the clean validation set is not involved in updating network parameters. To simulate different rates of noisy labels, we randomly flipped the label of a training set to other classes w.r.t. a given probability ϵ (5%, 10%, 20%, 30%, and 50%), which is also the proportion of the noisy labels.

During training, we extracted 40-dimensional Mel-frequency cepstral coefficients for each utterance. We applied the extended time delay network (E-TDNN) described in [24] for the x-vectors generation. The detailed implementations of E-TDNN are available on `https://github.com/Snowdar/asv-subtools`.

Each network was trained on NVIDIA 2080 RTX GPU with a mini-batch size of 512. The hyper-parameters in the label noise correction loss were set as $\alpha_T = 1$, $\gamma = 2$, $\beta = 1$, and $K = 3$ for the sub-center AM-Softmax, respectively. The learning rate of $1e{-}3$ was used and gradually reduced to $1e{-}6$ along with the AdamW optimizer, and it was trained for a total of 21 epochs. Since the model may overfit the wrong labels and result in final performance degradation, the model which achieved the highest validation set prediction accuracy was utilized to extract x-vectors. At the test stage, we adopted three different back-end scoring methods, namely, cosine distance, PLDA, and NL-PLDA, respectively. The performance was measured in terms of Equal Error Rate (EER). All of these systems were implemented on ASV-Subtools [25].

4.2. Evaluations and ablation studies on VoxCeleb1

In this section, we discuss our experiments and ablation studies on the VoxCeleb1 dataset. These studies help us to observe performance deteriorations in the presence of increasing noise and examine the effectiveness of label noise correction loss and sub-center AM-Softmax, respectively. All of these models were trained with different proportions of noisy labels and evaluated on the clean VoxCeleb1 test set.

As depicted in Figure 1, we first compared the model prediction accuracy between the baseline and our proposed method. We show the representative training processes with label noise proportions $\epsilon = 0\%$, 20%, and 50%. From Figure 1 (a), the convergence speed of clean training data is faster than noisy label data, which shows that the network takes longer to learn noisy samples. It also indicates that when the proportions of noisy labels increased to 20% and 50%, the network learns reasonable models at the beginning of training, as shown by the higher validation set prediction accuracy than the training set in the first few epochs. This is the basis for the success of our proposed approach. Unfortunately, the increasing number of training cycles leads to the neural network overfitting with respect to noisy samples, thereby degrading the validation set prediction accuracy of both two baseline networks. On the contrary, from Figure 1 (b), one sees that the proposed method suffers from fewer adverse effects due to mislabeled samples compared to the baseline, as shown by the fact that the model trained with label noise always produces a higher valid prediction accuracy in the training processes. Besides, it is quite remarkable that a final validation accuracy of around 90% can be attained in such a challenging task. We would like to emphasize that the final training accuracy of the proposed method is close to the expected true error ratio, which indicates that our proposed method separates correct labels and incorrect labels from a noisy dataset.

Table 1 summarizes the results of the first set of experiments. As expected, the E-TDNN baseline rapidly breaks down when noise ratios increase. Noisy labels severely deteriorate the performance of cosine scoring, while PLDA and NL-PLDA exhibit relative robustness to label noise. Table 1 indicates that using the sub-center AM-Softmax (Sub-AM) method could help achieve better performance, while adopting Label Noise Correction Loss (LNCL) significantly improves the performance in all noise ratios. The best results were achieved by combining the two components for front-end network training and adopting NL-PLDA for back-end scoring. The analyses show that each component in our proposed method is essential for the model to learn robustly.

Table 1: *EER(%) comparisons on VoxCeleb1*

System	Back-end	Proportion of Noisy Labels					
		0%	5%	10%	20%	30%	50%
Baseline	Cosine	4.88	6.90	7.19	8.56	11.81	16.02
	PLDA	3.92	5.46	5.75	6.51	7.25	8.73
	NL-PLDA	3.92	5.46	5.54	6.09	6.19	7.40
Sub-AM	Cosine	4.74	6.11	6.90	8.03	8.81	12.75
	PLDA	3.88	5.23	5.42	6.57	6.97	8.52
	NL-PLDA	**3.87**	5.22	5.32	5.94	6.10	7.24
LNCL	Cosine	4.66	4.99	5.52	5.73	6.42	7.63
	PLDA	3.98	4.67	5.29	6.08	6.70	8.29
	NL-PLDA	3.96	4.51	4.98	5.27	5.51	6.44
LNCL + Sub-AM	Cosine	4.64	4.83	5.11	5.45	5.56	6.32
	PLDA	3.92	4.45	5.02	5.47	5.96	7.29
	NL-PLDA	3.92	**4.32**	**4.63**	**4.85**	**5.04**	**5.65**

4.3. Evaluations on VoxCeleb1 data augmentation

One can see from the above that high prediction accuracy is a key factor to the success of our method, while hard samples may reduce the prediction accuracy. In this section, we discuss experiments that were conducted to evaluate our method in handling error labeled data with hard samples. To create hard samples, we artificially augmented the mislabeled VoxCeleb1 training set with noises contained in the MUSAN corpus [26]. Since noises result in the loss of speech intelligibility and quality, they reduce a speaker's discriminatory information [27, 28]. Samples with noise are difficult for networks to learn, so they can be considered as hard samples. After augmentation, the training set size is two times the original size. Evaluation results on the VoxCeleb1 test set are shown in Table 2. Comparisons between Table 2 and Table 1 indicate that data augmentation improves performance even with noisy labels. At the same time, the results indicate that hard samples do not impact the effectiveness of the proposed approach.

Table 2: *EER(%) comparisons on VoxCeleb1 augmentation*

System	Back-end	Proportion of Noisy Labels					
		0%	5%	10%	20%	30%	50%
Baseline	Cosine	4.01	5.01	5.42	6.81	7.66	13.28
	PLDA	3.51	4.52	4.67	6.09	6.55	8.30
	NL-PLDA	3.51	4.40	4.45	5.29	5.74	6.56
LNCL + Sub-AM	Cosine	4.00	4.22	4.42	5.12	5.39	6.17
	PLDA	**3.49**	3.90	4.12	5.14	5.75	6.93
	NL-PLDA	3.50	**3.86**	**3.87**	**4.39**	**4.59**	**5.19**

4.4. Evaluations on VoxCeleb2

In this section, we further evaluated the effectiveness of our proposed method on a larger dataset, VoxCeleb2. There are more speaker classes in this dataset, which can also cast negative impacts on the network prediction. We tested the performance on the VoxCeleb1 test set and the hard VoxCeleb1-H test set. The VoxCeleb1-H test set is a 'hard' evaluation set consisting of 552,536 pairs with 1,190 speakers who share the same nationality and gender. Table 3 presents the evaluation results. The proposed method shows robustness to larger data sets and demonstrates discriminability for hard pairs even with 50% error labels. Surprisingly, by adopting our proposed method, the cosine scoring outperforms NL-PLDA in lower noise ratios. This indicates that the data-driven nature of neural networks enables them to be robust against label noise when given a sufficient number of correct training samples.

Table 3: *EER(%) comparisons on VoxCeleb2*

System	Back-end	Proportion of Noisy Labels					
		0%	5%	10%	20%	30%	50%
VoxCeleb1 test set							
Baseline	Cosine	1.71	1.90	2.02	2.58	3.00	4.41
	PLDA	1.70	1.79	1.90	2.54	2.99	4.66
	NL-PLDA	1.71	1.73	1.82	2.14	2.58	3.42
LNCL + Sub-AM	Cosine	**1.63**	**1.63**	**1.64**	**1.69**	1.80	2.25
	PLDA	1.71	1.78	1.81	2.05	2.15	2.86
	NL-PLDA	1.70	1.72	1.73	1.75	**1.77**	**2.19**
VoxCeleb1-H test set							
Baseline	Cosine	3.04	3.20	3.46	4.07	5.09	7.45
	PLDA	3.06	3.18	3.25	4.05	5.08	7.75
	NL-PLDA	3.05	3.08	3.13	3.49	4.32	5.73
LNCL + Sub-AM	Cosine	**3.02**	**3.05**	**3.06**	3.26	**3.26**	3.77
	PLDA	3.03	3.10	3.15	3.57	3.76	4.76
	NL-PLDA	**3.02**	3.07	3.10	**3.20**	3.33	**3.58**

5. Conclusions

In this paper, we proposed a simple yet effective approach to deal with noisy labels in speaker verification tasks. Specifically, We corrected the original loss value by introducing the model's prediction labels into loss function and modifying the vanilla AM-Softmax. In the proposed method, sample labels are corrected on the fly based on the learned network's predictions. Finally, we investigated the robustness of three kinds of back-ends for handling noisy labels. Experiments conducted on the VoxCeleb1 and VoxCeleb2 datasets indicated the proposed approach showed superior performance under label noise conditions.

6. Acknowledgements

This research was funded by the National Natural Science Foundation of China (Grant No.61876160 and No.62001405).

7. References

[1] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[2] T. Zhou, Y. Zhao, and J. Wu, "ResNeXt and Res2Net structure for speaker verification," *arXiv preprint arXiv:2007.02480*, 2020.

[3] NIST, "NIST 2018 speaker recognition evaluation plan," 2018. [Online]. Available: https://www.nist.gov/system/files/documents/2018/08/17/sre18_eval_plan_2018-05-31_v6.pdf

[4] K. A. Lee, H. Yamamoto, K. Okabe, Q. Wang, L. Guo, T. Koshinaka, J. Zhang, and K. Shinoda, "NEC-TT system for mixed-bandwidth and multi-domain speaker recognition," *Computer Speech & Language*, vol. 61, p. 101033, 2020.

[5] B. J. Borgström and P. Torres-Carrasquillo, "Bayesian estimation of PLDA with noisy training labels, with applications to speaker verification," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7594–7598.

[6] J. Deng, J. Guo, T. Liu, M. Gong, and S. Zafeiriou, "Sub-center ArcFace: Boosting face recognition by large-scale noisy web faces," in *European Conference on Computer Vision*. Springer, 2020, pp. 741–757.

[7] R. Sachdeva, F. R. Cordeiro, V. Belagiannis, I. Reid, and G. Carneiro, "Evidentialmix: Learning with combined open-set and closed-set noisy labels," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 3607–3615.

[8] H. Song, M. Kim, D. Park, and J.-G. Lee, "Learning from noisy labels with deep neural networks: A survey," *arXiv preprint arXiv:2007.08199*, 2020.

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

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

[11] X. Liu, S. Li, M. Kan, S. Shan, and X. Chen, "Self-error-correcting convolutional neural network for learning with noisy labels," in *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*. IEEE, 2017, pp. 111–117.

[12] J. Yao, J. Wang, I. W. Tsang, Y. Zhang, J. Sun, C. Zhang, and R. Zhang, "Deep learning from noisy image labels with quality embedding," *IEEE Transactions on Image Processing*, vol. 28, no. 4, pp. 1909–1922, 2018.

[13] S. Jenni and P. Favaro, "Deep bilevel learning," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 618–633.

[14] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[15] R. Tanno, A. Saeedi, S. Sankaranarayanan, D. C. Alexander, and N. Silberman, "Learning from noisy labels by regularized estimation of annotator confusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 244–11 253.

[16] D. Hendrycks, K. Lee, and M. Mazeika, "Using pre-training can improve model robustness and uncertainty," in *International Conference on Machine Learning*. PMLR, 2019, pp. 2712–2721.

[17] Y. Lyu and I. W. Tsang, "Curriculum loss: Robust learning and generalization against label corruption," in *International Conference on Learning Representations*, 2019.

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

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

[20] S. E. Reed, H. Lee, D. Anguelov, C. Szegedy, D. Erhan, and A. Rabinovich, "Training deep neural networks on noisy labels with bootstrapping," in *ICLR (Workshop)*, 2015.

[21] P. Kenny, "Bayesian speaker verification with heavy-tailed priors." in *Odyssey*, vol. 14, 2010.

[22] S. Ioffe, "Probabilistic linear discriminant analysis," in *European Conference on Computer Vision*. Springer, 2006, pp. 531–542.

[23] D. Tanaka, D. Ikami, T. Yamasaki, and K. Aizawa, "Joint optimization framework for learning with noisy labels," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5552–5560.

[24] D. Snyder, D. Garcia-Romero, G. Sell, A. McCree, D. Povey, and S. Khudanpur, "Speaker recognition for multi-speaker conversations using x-vectors," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5796–5800.

[25] F. Tong, M. Zhao, J. Zhou, H. Lu, Z. Li, L. Li, and Q. Hong, "ASV-Subtools: Open source toolkit for automatic speaker verification," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6184–6188.

[26] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[27] D. Cai, W. Cai, and M. Li, "Within-sample variability-invariant loss for robust speaker recognition under noisy environments," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6469–6473.

[28] J. Zhou, T. Jiang, L. Li, Q. Hong, Z. Wang, and B. Xia, "Training multi-task adversarial network for extracting noise-robust speaker embedding," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6196–6200.