JEFFREYS DIVERGENCE-BASED REGULARIZATION OF NEURAL NETWORK OUTPUT DISTRIBUTION APPLIED TO SPEAKER RECOGNITION

Pierre-Michel Bousquet, Mickael Rouvier

JEFFREYS DIVERGENCE-BASED REGULARIZATION OF NEURAL NETWORK OUTPUT DISTRIBUTION APPLIED TO SPEAKER RECOGNITION

Pierre-Michel Bousquet, Mickael Rouvier

LIA - Avignon University

ABSTRACT

A new loss function for speaker recognition with deep neural network is proposed, based on Jeffreys Divergence. Adding this divergence to the cross-entropy loss function allows to maximize the target value of the output distribution while smoothing the non-target values. This objective function provides highly discriminative features. Beyond this effect, we propose a theoretical justification of its effectiveness and try to understand how this loss function affects the model, in particular the impact on dataset types (i.e. in-domain or out-of-domain w.r.t the training corpus). Our experiments show that Jeffreys loss consistently outperforms the state-of-the-art for speaker recognition, especially on out-of-domain data, and helps limit false alarms.

Index Terms— Speaker recognition, deep learning, loss function

1. INTRODUCTION

In recent years, Deep Neural Networks (DNN) have achieved remarkable performance in speaker recognition (SR) compared to the traditional i-vector/PLDA framework [1]. Proposing an original and discriminant voice representation, the DNN can be seen as a complex function that maps the audio to a vector (i.e. speaker embedding). The loss function plays an important role to determine the DNN parameters during the learning phase. Hence, choosing the suitable loss function is crucial to estimate discriminant parameters, achieve a good accuracy and avoid drawbacks that typically occur with deep learning (e.g. overfitting, miscalibration [2],...).

Two major dimensions of research around loss functions may be found in the machine learning literature. Some of them are based on classification (softmax cross-entropy loss, center loss [3]), while others achieve representation learning (contrastive loss [4], triplet loss [5, 6], circle loss [7], barlow twins [8, 9]). However, both types of loss functions suffer from major issues: the triplet loss for representation learning, for instance, exhibits a combinatorial explosion in the number of possible triplets, especially for large-scale datasets, leading to a drastically increased number of training steps. On the other hand, loss functions based on classification may see a linear increase of the size of the linear transformation matrix with the number of identities; the learned features are separable for the closed-set classification problem but not discriminative enough for the open-set SR problem.

The softmax cross-entropy loss is typically good for optimizing the inter-class difference (i.e., separating different classes) but not for reducing the intra-class variation (i.e., making classes more compact). To address this issue, many loss functions have been proposed, attempting to minimize the intra-class variation: sphereFace [10], cosFace [11, 12], arcFace [13]. Based on angular distances, which tend to be the state of the art in SR, they can be seen as *embedding losses* as they all rely on the generic softmax loss function. Therefore, completing the cross-entropy loss by a regularizer of the output distribution could improve all these configurations.

Two goals can be identified for the objective function of a DNN: reflect the true objective of the model learning (to discriminate the training speakers) and the real objective of the system (to avoid overfitting, in order to generalize well to new data). These two goals are independent, even contradictory, and the regularizer will have to find a trade-off between them.

In this paper, a new regularizer of output distributions, the Jeffreys loss, is presented in Section 4. Before that, Section 2 and 3 present the different loss functions, probe the output distribution and justify our approach. Results of wide and deep ResNet systems on speaker verification tasks are analyzed in Section 5 and conclusions are provided in Section 6.

2. LOSS FUNCTIONS FOR SPEAKER RECOGNITION

Regularizing the output distribution of deep and wide neural networks has long been unexplored. Since, many studies have shown the benefits of loss function regularizers. First recall that, for each training example, the model computes a conditional distribution over labels $k \in \{1...K\}$ given the x-vector x through a softmax function: $p(k|x) = \exp(z_k) / \sum_{i=1}^{K} \exp(z_i)$ where z_i are the logits. In SR, logits are actually dot product [14] or cosine (coupled with the angular softmax loss function [11, 13]), eventually shifted and scaled (penalty margin [11], temperature scaling). The most currently used loss in SR is the cross-entropy in the case of hard target (a single ground-truth label equal to 1 for k, otherwise 0) and minimizing the cross entropy is equivalent to maximizing the logit of the correct label. Omitting the dependence of p on example x and denoting $p(k|x)$ by p_k, the cross-entropy loss for the example below is equal to $\mathcal{L}_{CE} = -\log(p_k)$.

In SR, learning enhancement by temperature scaling of the logit is typically set to low values, fastening convergence and limiting overfitting. It has been also shown that it provides better calibrated scores [2]. A popular countermeasure against overfitting is addition of a regularization term to the objective function. Label-smoothing [15], widespread in many fields (image classification, language modeling, machine translation, speech recognition, digit recognition, ...), replaces a "hard" target label objective by a "softened" one, playing on non-target values of the output distribution. This leads to add to $\mathcal{L}_{CE}$ a weighted term $\mathcal{L}_{LS} = \frac{1}{K-1} \sum_{i \neq k} \log p_i$. Some variants have been proposed [16, 17]. Smoothing the labels by other ways, such as virtual adversarial training [18], adding label noise [19, 20] or addressing class imbalance [21], has also been effective in preventing overfitting and, thus, improving generalization. Each time, cooperation of these methods with weight decay [22] must be analyzed and overcome.

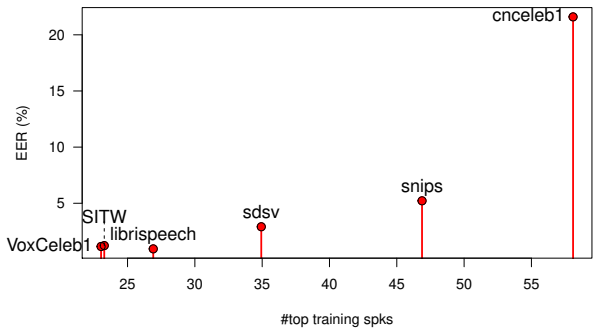

Fig. 1. Equal error rates on the six evaluations presented in [23] as a function of the average number of top training speakers (x-values). See Section 3 and [23] for more details.

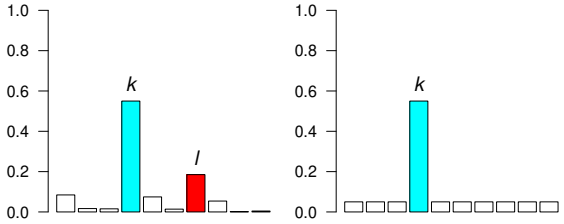

Fig. 2. Two examples of output softmax distribution for an example of the k^{th} training speaker. As their k^{th} values are equal, both yield the same value of one-hot-target cross-entropy loss.

In the following, the benefits of the output regularizing and its ability to better achieve the two goals defined in the introduction are analyzed and justified, more thoroughly than in the literature, leading us to propose a more comprehensive regularizer of the output distribution for SR.

3. PROBING THE OUTPUT DISTRIBUTIONS

3.1. Better discriminate the training speakers

In [23] the notion of *top training speakers* is introduced, that is, given an utterance of a speaker unknown to the system, the dominant labels of its output distribution. To summarize the approach, once the x-vector of a test utterance is extracted, the softmax of the last layer is computed and the labels with the highest values are retained. The corresponding training speakers are those who are the most involved in the modeling of this utterance.

To take this study a step further, Figure 1 reports, for each evaluation presented in [23], the average number of top training speakers (x-values) of its utterance set and the equal error rates (EER) computed on its trial set (y-values). From Figure 1, it is clear that an increase in the number of top training speakers correlates with an increase in the observed EER. The network, trained as a classifier for the training set, proceeds with new speakers by similarity to it. Too many top-speakers reveal some difficulties in the system to model the data and, therefore, the SR system performs better when it succeeds in modeling test data with a low number of main training speakers. This shows that performance of a system on a domain (and similarity of the latter with the training data) can be predicted by only probing the last layer of the data, but it also highlights that the ability of a system to generalize is measurable by the entropy between test and training output distributions.

Figure 2 illustrates the link with the entropy between the training data outputs. The figure shows two examples of output softmax distribution for an example of the k^{th} training speaker (the training speaker sample size is limited for readability). As expected, the target value p_k is maximal for both cases, and equal, so that the cross-entropy loss induced by both examples is identical.

However, on the left, a non-negligible "foreign" (non-target) value for the l^{th} label (red bar) reduces the entropy between the output distributions of the k^{th} and l^{th} training speakers (which can be measured by the symmetric Kullback-Leibler divergence). This concern is not taken into account by the cross-entropy loss. The distribution on the right of the Figure, where the foreign values $\{p_i\}_{i\neq k}$ follow a uniform distribution, increases the entropy between the output distributions of training speakers k and l, thus helping to better discriminate them and to achieve the first goal outlined in the previous section. Techniques such as label-smoothing rely on this observation and attempt to equalize the non-target values of the output.

3.2. Avoid overfitting

But label-smoothing could lead to overfitting, by taking too much into account the specificity of the training data. This finding is contrary to what is usually claimed and empirically justified [24, 16].

In what follows, we propose to better explain why smoothing the non-target labels also respects the second goal of the learning phase: avoid overfitting. Let q_1 and q_2 denote the two output distributions displayed in Figure 2. Now consider an SR domain and its output set $\mathcal{P}$. The mean entropy between $\mathcal{P}$ and q_1 or q_2 can be estimated by the expectations $\mathbf{E}_{p\in\mathcal{P}}\left[D_{KL}\left(p||q_i\right)\right]$, $i = 1, 2$. This amounts to comparing $\mathbf{E}_{p\in\mathcal{P}}\left[p\right].\log q_i$ where '.' denotes the dot product. When $\mathcal{P}$ is far from the training domain, the values of p tend to be spread across many labels (as observed above and in Figure 1), so that $\mathbf{E}_{p\in\mathcal{P}}\left[p\right]$ tends towards the uniform distribution, many top speakers inducing smoother output. Therefore, $\mathbf{E}_{p\in\mathcal{P}}\left[D_{KL}\left(p||q_2\right)\right]$ should probably be lower than $\mathbf{E}_{p\in\mathcal{P}}\left[D_{KL}\left(p||q_1\right)\right]$.

In other words, by smoothing the non-target values of the training set output distributions, redundancy between training speakers is reduced but, also, the entropy of out-of-domain data a posteriori of the model.

4. JEFFREYS-BASED LOSS FUNCTION

As shown above, regularizing the network w.r.t. the two goals defined in Section 1 can be done by moving the non-target output distribution p closer to the uniform distribution u. The most complete entropy measure between p and u is the symmetric Kullback-Leibler divergence (also referred to as Jeffreys divergence). This divergence takes into account the entropy of u a posteriori of p (as done in label-smoothing) but, above all, the one of p a posteriori of u.

To apply Jeffreys divergence on the non-target output softmax values $[p_i]_{i\neq k}$, this sub-vector is $L1$-normalized to become a distribution (thus divided by $1-p_k$), then the Jeffreys divergence-based loss between it and the uniform distribution u, equal to $\frac{1}{K-1}$ for all its $K-1$ values, is computed:

$$\mathcal{L}_J = D_{KL}\left(u|\left[\frac{p_i}{1-p_k}\right]_{i\neq k}\right) + D_{KL}\left(\left[\frac{p_i}{1-p_k}\right]_{i\neq k}|u\right) \quad (1)$$

After simplification, this is equal to :

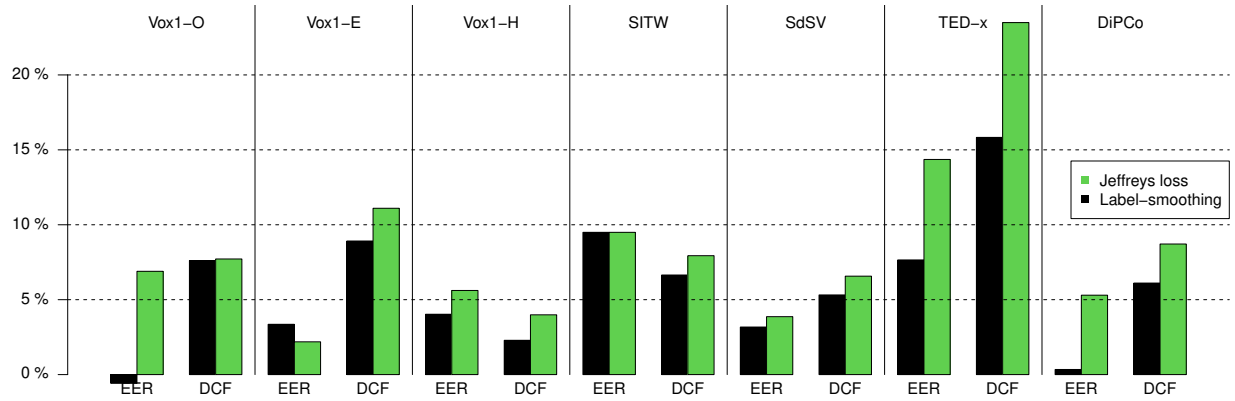

Fig. 3. Relative gains between the 'hard'-target cross entropy loss (row 1 of Table 1) and two regularizers: label-smoothing (black bars and row 3 of Table 1) then Jeffreys loss (green bars, row 4 of the Table) in terms of EER and DCF. The VoxCeleb1 evaluations are fully in-domain, SITW is relatively in-domain, the others are out-of-domain.

Table 1. Comparison of the cross entropy loss with two regularizers: label-smoothing (with or without weight decay) and the new Jeffreys-based loss function.

System	**VoxCeleb1** *-O Cleaned*		**VoxCeleb1** *-E Cleaned*		**VoxCeleb1** *-H Cleaned*		**SITW** *core-core*		**SdSV** *task 2*		**TED-x Spanish**		**DiPCo**	
	EER	DCF	EER	DCF	EER	DCF	EER	DCF	EER	DCF	EER	DCF	EER	DCF
AAM-Softmax	0.93	0.095	1.01	0.114	1.76	0.170	1.15	0.101	3.46	0.295	2.33	0.178	5.85	0.368
AAM-Softmax + LS with weigth-decay	0.94	0.087	1.05	0.112	1.80	0.170	1.15	0.101	3.63	0.296	2.26	0.140	5.60	0.356
AAM-Softmax + LS w/o weigth-decay	0.93	0.087	0.98	0.104	1.69	0.166	1.04	0.094	3.36	0.280	2.15	0.149	5.83	0.346
AAM-Jeffreys	0.86	0.087	0.99	0.102	1.66	0.164	1.04	0.093	3.33	0.276	1.99	0.136	5.54	0.336

$$\mathcal{L}_J = -\frac{\sum_{i\neq k}\log p_i}{K-1} + \frac{\sum_{i\neq k} p_i \log p_i}{1-p_k} \tag{2}$$

The final loss is a weighted sum of the cross-entropy and Jeffreys losses :

$$\mathcal{L} = \mathcal{L}_{CE} + \alpha\mathcal{L}_J \tag{3}$$

where α is a scalar. The second term of Equation 2 is a little bit 'hard' as it inserts p_k instead of $\log(p_k)$ inside the loss function. To alleviate this effect, the two terms of Jeffreys loss are independently weighted [1]:

$$\mathcal{L} = -\log(p_k) - \alpha\frac{\sum_{i\neq k}\log p_i}{K-1} + \beta\frac{\sum_{i\neq k} p_i \log p_i}{1-p_k} \tag{4}$$

This loss can be rewritten by using the label-smoothing loss:

$$\mathcal{L} = \mathcal{L}_{CE} + \alpha\mathcal{L}_{LS} + \beta\frac{\sum_{i\neq k} p_i \log p_i}{1-p_k} \tag{5}$$

This result shows that label-smoothing is only a part of the divergence between non-target values and uniform distribution. The last term of Equation 5 simultaneously forces the non-target values to be as uniform as possible and completes the cross-entropy loss objective (thanks to the denominator $1-p_k$) [2].

[1] To facilitate further research, the code of the loss-function is available on https://github.com/mrouvier/jeffreys_loss

[2] Let us note that [16] includes a term $p_i \log p_i$ in a loss function, but without the denominator, including the target value and which does not improve performance when combined with label-smoothing.

5. EXPERIMENTS

5.1. Experimental setup

The x-vector extractor used in this paper is a variant based on ResNet-34. The extractor was trained on the development part of the Voxceleb 2 dataset [25], cut into 4-second chunks and augmented with noise, as described in [14] and available as a part of the Kaldi-recipe. It contains about 1M segments (+ 4M augmented) of 5994 speakers. As input, we used 60-dimensional filter-banks. The speaker embeddings are 256-dimensional and the loss is the angular additive margin with temperature scaling equal to 30 and margin equal to 0.2. The sizes of the feature maps are 128, 128, 256 and 256 for the four ResNet blocks. We use stochastic gradient descent with momentum equal to 0.9, a weight decay equal to 2.10^{-4} and initial learning rate equal to 0.2. The implementation is based on PyTorch. For scoring, the x-vectors are centered by subtracting the overall mean of the training dataset, then the cosine metric is applied. Despite the shift between some tests and training data, no domain adaptation technique is performed in order to fairly compare the effects of regularizers.

The relevance of the methods is tested on seven datasets: VoxCeleb1-O, E and H (cleaned versions) [26, 27], Speakers In The Wild (SITW) core-core task [28], the Short duration Speaker Verification (SdSV) challenge Task 2 (a text-independent SR evaluation based on the DeepMine dataset [29, 30], comprised of Persian-native and some English-non native utterances), DiPCo [31] (a far-field speaker verification corpus issued from DiPCo corpus) and TED-x Spanish. The latter is derived from the public-available TED-x Spanish dataset, created from TED talks in Spanish and aiming to be used in the Automatic Speech Recognition Task. For the

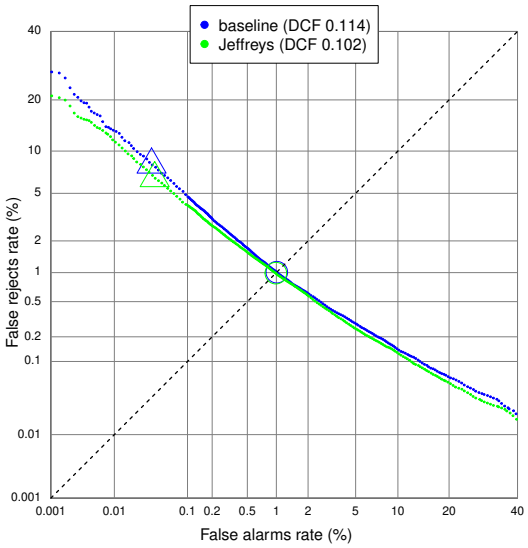

Fig. 4. DET curves for VoxCeleb1-E. The circles are the EERs, the triangles the DCFs.

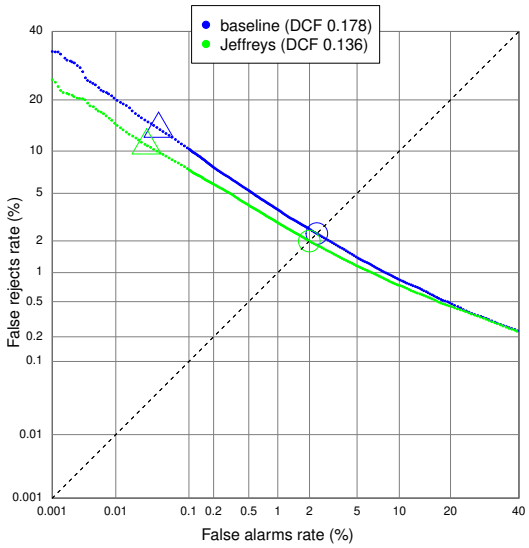

Fig. 5. DET curves for TED-x Spanish.

derived corpus, the segments have a duration range between 3 and 10 seconds and we randomly selected 2M pairs (1.6M non-targets pairs and 0.4M targets pairs), all with the same gender.

The last three evaluations are "out-of-domain", due to mismatch of language or recording conditions.

The results are reported in terms of Equal Error-Rate (EER) and normalized minimal detection cost (DCF) with the probability of a target trial set to 0.01 and the cost of miss-detection and false alarm set to 1.

5.2. Results

Table 1 shows the evaluation results. Row 1 reports results of the system learned with the cross-entropy loss function. This system is stated here as the baseline. Row 2 reports results with label-smoothing. As weight decay is also a regularization technique and could interfere and clash with label-smoothing, row 3 reports results obtained without weight decay. Comparing the first three rows show that label-smoothing improves performance provided that weight decay is disabled. Let us note that the best weight for label-smoothing, in terms of performance, was equal to $\alpha = 0.1$. Row 4 reports results with the proposed Jeffreys-based loss function. No weight decay is applied. The best weights of Eq. 4 were $\alpha = 0.1$ and $\beta = 0.025$.

To more easily assess the benefits of the regularizers, Figure 3 visualizes the relative gain between the hard-target cross entropy and the two approaches: label-smoothing (black bars), then Jeffreys loss (green bars). On VoxCeleb1 evaluations, which are fully in-domain, the gains of performance confirm the ability of the non-target label smoothings to better fulfill the first goal ("*to discriminate the training speakers*"). These gains are comparable to those observed in other fields [15, 16] and sometimes even greater. The gain on SITW is significant with both methods. On out-of-domain evaluations (SdSV, TED-x Spanish, DiPCo), the significant gains of performance demonstrate that the regularizers also achieve the second goal ("*to generalize well*").

The new Jeffreys-based loss function always yields better accuracy than label-smoothing (except for VoxCeleb 1-E EER), especially for out-of-domain evaluations. The proposed approach tackles overfitting and can be considered more robust to domain mismatch than label-smoothing. In particular, the Jeffreys regularizer provides significant gains in terms of DCF, even spectacular for out-of-domain evaluations. The method helps to regularize the upper tail of the non target score distribution, as illustrated by detection error tradeoff (DET) plots in Figures 4 and 5. This ability to produce more calibrated scores is of the utmost importance for critical applications (forensic, security) where false alarms must be severely penalized.

6. CONCLUSION

In this work, several topics about regularization of SR neural network outputs are addressed. First, label-smoothing deserved to be tested for SR. Its results are reported, tested on various evaluations more or less far from the training domain. Second, this regularization is claimed to avoid overfitting, but this outcome is only checked empirically. Here, we show why this assertion is paradoxical a priori, and propose a more theoretical justification showing how such soft target approaches can simultaneously achieve both objectives defined above: discriminating the training speakers and generalizing well to new data. These investigations lead us to propose a new loss function for SR DNN, more comprehensive and robust than label-smoothing, which improves accuracy of the recognition, in particular for the cases of domain mismatch and critical applications. Moreover, this new loss function is compatible with all recent techniques used in SR: sphereFace, cosFace, arcFace...

These results seem promising enough to propose future work testing this new loss function on other fields (image classification, machine translation, language modeling, speech recognition) on which label-smoothing has proven to be effective.

7. ACKNOWLEDGEMENTS

This work was supported by the VoicePersonae project ANR-18-JSTS-0001, granted access to the HPC resources of IDRIS under the allocation 2022-AD011013257R1 made by GENCI and supported by the ANR agency (Agence Nationale de la Recherche). Many thanks to Teva Merlin for proofreading this work.

8. REFERENCES

[1] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis for Speaker Verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[2] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proceedings of the 34th International Conference on Machine Learning - Volume 70*. JMLR.org, 2017, pp. 1321–1330.

[3] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *Computer Vision – ECCV 2016*. Springer, 2016, pp. 499–515.

[4] G. Heigold, I. Moreno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," in *ICASSP*, 2016, pp. 5115–5119.

[5] C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, and Z. Zhu, "Deep speaker: an end-to-end neural speaker embedding system," *arXiv preprint arXiv:1705.02304*, 2017.

[6] H. Bredin, "Tristounet: triplet loss for speaker turn embedding," in *ICASSP*, 2017, pp. 5430–5434.

[7] Y. Sun, C. Cheng, Y. Zhang, C. Zhang, L. Zheng, Z. Wang, and Y. Wei, "Circle loss: A unified perspective of pair similarity optimization," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6397–6406, 2020.

[8] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow twins: Self-supervised learning via redundancy reduction," in *International Conference on Machine Learning, ICML*, vol. 139, 2021, pp. 12 310–12 320.

[9] M. Mohammadamini, D. Matrouf, J.-F. Bonastre, S. Dowerah, R. Serizel, and D. Jouvet, "Barlow Twins self-supervised learning for robust speaker recognition," in *Interspeech*, 2022.

[10] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," *CoRR*, vol. abs/1704.08063, 2017.

[11] F. Wang, J. Cheng, W. Liu, and H. Liu, "Additive margin softmax for face verification," *IEEE Signal Processing Letters*, vol. 25, no. 7, pp. 926–930, 2018.

[12] H. Wang, Y. Wang, Z. Zhou, X. Ji, Z. Li, D. Gong, J. Zhou, and W. Liu, "Cosface: Large margin cosine loss for deep face recognition," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5265–5274, 2018.

[13] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," 2019.

[14] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *ICASSP*. IEEE, 2018, pp. 5329–5333.

[15] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2818–2826.

[16] G. Pereyra, G. Tucker, J. Chorowski, Ł. Kaiser, and G. Hinton, "Regularizing neural networks by penalizing confident output distributions," *arXiv preprint arXiv:1701.06548*, 2017.

[17] J. Lienen and E. Hüllermeier, "From label smoothing to label relaxation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 10, 2021, pp. 8583–8591.

[18] T. Miyato, S. Maeda, M. Koyama, K. Nakae, and S. Ishii, "Distributional smoothing by virtual adversarial examples," in *ICLR (Poster)*, 2016.

[19] S. Reed, H. Lee, D. Anguelov, C. Szegedy, D. Erhan, and A. Rabinovich, "Training deep neural networks on noisy labels with bootstrapping," *ArXiv*, vol. abs/1412.6596, 2014.

[20] L. Xie, J. Wang, Z. Wei, M. Wang, and Q. Tian, "DisturbLabel: Regularizing CNN on the loss layer," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4753–4762.

[21] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017.

[22] A. Krizhevsky, "Learning multiple layers of features from tiny images," University of Toronto, Tech. Rep., 2009.

[23] P.-M. Bousquet, M. Rouvier, and J.-F. Bonastre, "Reliability criterion based on learning-phase entropy for speaker recognition with neural network," in *Proc. Interspeech*, 2022.

[24] R. Muller, S. Kornblith, and G. Hinton, "When does label smoothing help?" *ArXiv*, vol. abs/1906.02629, 2019.

[25] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *Interspeech*, Sep 2018.

[26] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: A large-scale speaker identification dataset," *Interspeech*, Aug 2017.

[27] W. Xie, A. Nagrani, J. S. Chung, and A. Zisserman, "Utterance-level aggregation for speaker recognition in the wild," in *ICASSP*, 2019, pp. 5791–5795.

[28] M. McLaren, L. Ferrer, D. Castán, and A. D. Lawson, "The 2016 Speakers in the Wild speaker recognition evaluation," in *Interspeech*, 2016, pp. 818–822.

[29] H. Zeinali, H. Sameti, and T. Stafylakis, "DeepMine speech processing database: Text-dependent and independent speaker verification and speech recognition in Persian and English." in *Speaker and Language Recognition Workshop (IEEE Odyssey)*, 2018, pp. 386–392.

[30] H. Zeinali, L. Burget, and J. Cernocky, "A multi purpose and large scale speech corpus in Persian and English for speaker and speech recognition: the DeepMine database," in *Proc. ASRU 2019 The 2019 IEEE Automatic Speech Recognition and Understanding Workshop*, 2019.

[31] M. Rouvier and M. Mohammadamini, "Far-field speaker recognition benchmark derived from the DiPCo corpus," in *Preprint : Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'22)*, 2022.