ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9414261

SELF-SUPERVISED LEARNING BASED DOMAIN ADAPTATION FOR ROBUST SPEAKER VERIFICATION

Zhengyang Chen, Shuai Wang, Yanmin Qian[†]

MoE Key Lab of Artificial Intelligence, AI Institute
SpeechLab, Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai, China
{zhengyang.chen, feixiang121976, yanminqian}@sjtu.edu.cn

ABSTRACT

Large performance degradation is often observed for speaker verification systems when applied to a new domain dataset. Given an unlabeled target-domain dataset, unsupervised domain adaptation (UDA) methods, which usually leverage adversarial training strategies, are commonly used to bridge the performance gap caused by the domain mismatch. However, such adversarial training strategy only uses the distribution information of target domain data and can not ensure the performance improvement on the target domain. In this paper, we incorporate self-supervised learning strategy to the unsupervised domain adaptation system and proposed a self-supervised learning based domain adaptation approach (SSDA). Compared to the traditional UDA method, the new SSDA training strategy can fully leverage the potential label information from target domain and adapt the speaker discrimination ability from source domain simultaneously. We evaluated the proposed approach on the VoxCeleb (labeled source domain) and CnCeleb (unlabeled target domain) datasets, and the best SSDA system obtains 10.2% Equal Error Rate (EER) on the CnCeleb dataset without using any speaker labels on CnCeleb, which also can achieve the state-of-the-art results on this corpus.

Index Terms— Domain Adaptation, Self-Supervised Learning, Speaker Verification, Contrastive Learning

1. INTRODUCTION

Speaker verification aims to verify a person's identity given his or her voice. In recent years, the thriving of deep neural network (DNN) has led to great success of speaker verification systems [1]. To improve the performance and robustness of speaker verification systems, researchers have designed different network backbones [2, 3, 4], different pooling functions [5, 6, 7] and loss functions [8, 9, 10, 11, 12].

However, DNN based speaker verification systems usually require a large amount of well-labeled data for training, which is not available in most cases. On the other hand, a well-trained speaker verification system suffers from severe performance degradation when adapted to another dataset from a different domain. Thus, it is necessary to develop a method to fast adapt an existing model trained on well-labeled source domain data to a new target domain dataset where no speaker label is available. Such a task is considered as unsupervised domain adaptation (UDA) [13, 14].

Researchers have proposed different UDA methods to tackle this problem. The most common practice is using adversarial training strategy [15, 16, 17, 18] to minimize the distribution mismatch of the learned embeddings from different domains, which is expected to maintain the speaker discrimination ability learned from the well-labeled source domain data to the target domain. Other researchers have tried to use clustering methods [19] to estimate pseudo-labels for unlabeled target domain data and then do supervised training using the estimated labels. However, in adversarial training-based UDA methods, only the data distribution information from a different domain is used, and too aggressively matching the embedding distribution from different domains may hurt the speaker discrimination ability to some extent [18]. Besides, in clustering-based UDA methods, it is hard to determine the speaker number when doing clustering and the estimated label may not be accurate.

To fully leverage the available information from the target domain dataset, we proposed a self-supervised learning based domain adaptation method (SSDA) for unsupervised domain adaptation. According to the continuity of speech, there is usually only one person speaking in short time duration. Here, we assume that each utterance only contains one speaker, which is true for many datasets[1]. We can then sample positive pairs from the same utterance and sample negative pairs from different utterances to do contrastive learning [20] without reaching the speaker labels. Although purely self-supervised learning could be done with an unlabeled target domain dataset, the performance is usually not guaranteed. In this paper, we apply this self-supervised learning strategy to adapt a well-trained source speaker embedding extractor to a target domain dataset, simultaneously taking advantage of the rich information in both the source and target datasets. Experiments are carried out on the VoxCeleb and CnCeleb dataset. Our best SSDA system achieves 10.2% EER on the CnCeleb dataset even without using any speaker label and exceeds the current state-of-the-art result. Besides, when speaker labels from CnCeleb are used and source and target domain data are jointly trained, the system makes further improvement and achieves 8.86% EER on the CnCeleb evaluation set.

2. METHOD

This section will first give a brief introduction to the contrastive learning strategy, which is the foundation of self-supervised training. Then, four objective functions for constrastive learning will be introduced, and we will implement and compare their effects for our SSDA experiments in the following sections. Finally, we will present the self-supervised adaptation (SSDA) algorithm, together with the system configuration and training strategies.

[†]Yanmin Qian is the corresponding author

[1]Even for those conversational speeches, we still can easily prepare the data by using techniques such as speaker diarization

2.1. Contrastive Learning

Contrastive learning aims to maximize the similarity of positive pairs and minimize the similarity of negative pairs. Recently, many researchers have been studying to construct contrastive pairs based on an unlabeled dataset to do self-supervised training. For computer visual representation learning, researchers have used augmentation methods to construct positive and negative pairs for contrastive learning [21, 20] and any label is not needed here. Because of the continuity of speech, there is always only one person speaking in short time duration. Thus, for speaker representation learning, there is no need to do data augmentation for positive pair sampling and we can randomly sample two segments from one utterance and consider them as a positive pair. Similarly, we can consider two segments sampled from different utterances as a negative pair.

2.2. Contrastive Learning Objectives for Self-Supervised Training

For contrastive learning objectives, following the implementation in [22], we randomly sample M segments from each of N utterances, whose embeddings are $x_{j,i}$ where $1 \leq j \leq N$ and $1 \leq i \leq M$. The segments sampled from the same utterance are considered from the same speaker and segments from different utterance are considered from different speaker.

2.2.1. Contrastive Loss

The contrastive loss aims to maximize the distance of negative pairs and minimize the positive pairs' distance in a mini-batch. As shown in equation 1, positive pairs are sampled from the same utterance and negative pairs are sampled from different utterances within the mini-batch based on a hard negative mining strategy, which requires $M = 2$ in this condition. The margin m is set to 4.0 here.

$$\begin{aligned}\mathcal{L}_C = &\frac{1}{N}\sum_{j=1}^{N}\|\mathbf{x}_{j,1}-\mathbf{x}_{j,2}\|_2^2 \\ &+\frac{1}{N}\sum_{j=1}^{N}\max(0, m-\|\mathbf{x}_{j,1}-\mathbf{x}_{k\neq j,2}\|_2^2)\end{aligned} \tag{1}$$

2.2.2. Triplet Loss

Triplet loss minimizes the L2 distance between an anchor and a positive, and maximizes the distance between an anchor and a negative. As shown in equation 2, x_k is also sampled using a hard negative mining strategy as the last section and M is set to 2. The margin m is set to 4.0 here.

$$L_{\mathrm{T}} = \frac{1}{N}\sum_{j=1}^{N}\max\left(0, \|\mathbf{x}_{j,1}-\mathbf{x}_{j,2}\|_2^2 - \|\mathbf{x}_{j,1}-\mathbf{x}_{k\neq j,2}\|_2^2 + m\right) \tag{2}$$

2.2.3. Angular Prototypical Loss

Prototypical loss can also be considered as a kind of contrastive learning objectives, where the pair is constructed between a centriod and a query. For prototypical loss (ProtoLoss), each mini-batch contains a support set S and a query set Q. Same as the implementation in [22], the M-th segment from each utterance is considered as query. Then the prototype (centroid) is defined as:

$$\mathbf{c}_j = \frac{1}{M-1}\sum_{m=1}^{M-1}\mathbf{x}_{j,m} \tag{3}$$

Here, we use the angular version prototypical loss and cosine score is used to measure the similarity. Different from the original format in [22], we add a temperature hyper-parameter τ to the cosine similarity which controls the concentration level of the distribution following [23] and the score is defined as:

$$\mathbf{S}_{j,k} = \tau \cdot \cos\left(\mathbf{x}_{j,M}, \mathbf{c}_k\right) \tag{4}$$

The angular prototypical loss is calculated by a softmax function, in which each query is classified against N centroids:

$$L_{\mathrm{P}} = -\frac{1}{N}\sum_{j=1}^{N}\log\frac{e^{\mathbf{S}_{j,j}}}{\sum_{k=1}^{N}e^{\mathbf{S}_{j,k}}} \tag{5}$$

2.2.4. Generalised End to End Loss (GE2E)

Different from prototypical loss, in GE2E, each segment in a mini-batch is used to calculate the centroids:

$$\mathbf{c}_j = \frac{1}{M}\sum_{m=1}^{M}\mathbf{x}_{j,m} \tag{6}$$

$$\mathbf{c}_j^{(-i)} = \frac{1}{M-1}\sum_{\substack{m=1\\m\neq i}}^{M}\mathbf{x}_{j,m} \tag{7}$$

Here, each segment is also used as a query. When query and the centroid are from the same utterance, the query itself is excluded when calculating the centroid. We also add a temperature hyper-parameter τ to the cosine similarity:

$$\mathbf{S}_{j,i,k} = \begin{cases}\tau\cdot\cos\left(\mathbf{x}_{j,i}, \mathbf{c}_j^{(-i)}\right) & \text{if} \quad k=j \\ \tau\cdot\cos\left(\mathbf{x}_{j,i}, \mathbf{c}_k\right) & \text{otherwise}\end{cases} \tag{8}$$

The GE2E loss is defined as:

$$L_{\mathrm{G}} = -\frac{1}{N\cdot M}\sum_{j,i}\log\frac{e^{\mathbf{S}_{j,i,j}}}{\sum_{k=1}^{N}e^{\mathbf{S}_{j,i,k}}} \tag{9}$$

2.3. Domain Adaptation with Self-Supervised Learning

2.3.1. Self-Supervised adaptation training

In this section, we first introduce a simple self-supervised learning based domain adaptation (**SSDA**) training strategy. To adapt the model from the source domain, we first initialized the embedding extractor by the well-trained model from the source domain. Then, we do self-supervised training on the target domain data based on the contrastive learning objectives in section 2.2.

As shown in the solid line part of Figure 1, the embedding extractor is only supervised by the contrastive learning loss L_{cl} during this adaptation.

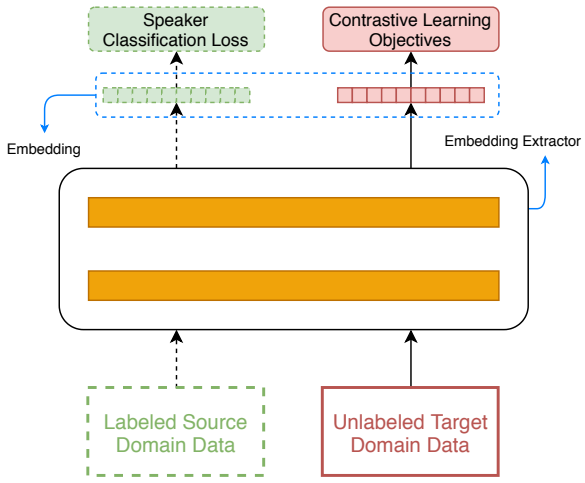

Fig. 1: Self-Supervised learning based domain adaptation.

2.3.2. *Self-Supervised adaptation with joint training*

Simply using the self-supervised learning strategy may cause the overfitting on the target domain data and the embedding extractor may lose the strong speaker discrimination ability, which benefits from source domain pre-training. Here, we proposed a joint training strategy for the SSDA, and it is named as **SSDA-Joint**.

As shown in Figure 1, the labeled source domain data and unlabeled target domain data are fed into the embedding extractor simultaneously during the training stage. The output source domain embeddings are supervised by the speaker classification loss and target domain embeddings are supervised by the contrastive learning loss. These two losses are jointly optimized and the total loss is defined as:

$$L_{total} = L_{cla} + \lambda L_{cl} \tag{10}$$

where L_{cla} is the speaker classification loss and λ is a hyper-parameter for weighted summation.

3. EXPERIMENTAL SETUP

3.1. Dataset

To perform domain adaptation, we use two datasets from different languages in our experiments and the detailed information of training parts for these two datasets is shown in Table 1. In our experiment, the data from VoxCeleb1 & 2 [24] is considered as source domain data. The source domain data is downloaded from Youtube and most of the speech data is in English. Besides, the data from CnCeleb [25] is considered as target domain data. The target domain data is downloaded from Bilibili and the speech data is in Chinese.

The 5994 speakers from the VoxCeleb2 dev set are used as source domain training data and the whole VoxCeleb1 is used as source domain evaluation set. For target domain data, following the official split, 800 speakers are used for training and 200 speakers are used for evaluation. It should be noted that we don't use any speaker label when doing target domain self-supervised learning.

We use the official trial list for CnCeleb to evaluate our system. This trial list contains 18,024 target pairs and 3,586,776 non-target pairs. Besides, when we evaluate our system on the source domain dataset, we consider the whole Voxceleb1 set as the evaluation dataset and all three official trial lists Vox1-O, Vox1-E and Vox1-H are used for evaluation.

Table 1: **Training data information.** The CnCeleb training utterance number is the statistic after preprocessing which is described section 3.2.

	Source Domain	Target Domain
Data	VoxCeleb	CnCeleb
Language	Mostly English	Chinese
# of Spk	5,994	800
# of Utt	1,021,161	53,288
# of Hour	2,207	148

3.2. Data Preprocessing

For CnCeleb training data, we first combine the short utterances to make them longer than 5 seconds because there are too many very short utterances. After processing, there are 53288 utterances left. For all the audio data, 40-dimensional Fbank features are extracted using Kaldi toolkit [26] with a 25ms window and 10ms frame shift, and silence is removed using an energy-based voice activity detector. Then we do the cepstral mean on the Fbank features with a sliding-window size of 300 frames. Similar to the Kaldi VoxCeleb recipe, we also discard all the utterances of less than 400 frames.

3.3. System Configuration

Resnet based r-vector [4] is used as the embedding extractor in our experiment and the embedding dimension is set to 256. In our experiments, we find it is better to set $M = 2$ for ProtoLoss and GE2E and we set the temperature hyper-parameter τ to 32. When additive angular margin (AAM) loss [9] is used in our experiment, margin is set to 0.2 for source domain data and 0.15 for target domain data. For simplicity, we set $\lambda = 1$ in equation 10 for SSDA-Joint training loss. Besides, the pre-trained model from source domain is also trained using AAM loss with the same configuration. For all the experiments in this paper, we use cosine similarity to score the trials.

4. RESULTS

4.1. Comparison between Supervised and Self-Supervised Baselines.

Table 2: **Comparison between supervised and self-supervised results on the CnCeleb.**

Train Data	Train Mode	Loss	EER (%)
VoxCeleb*	Supervised	AAM	12.11
CnCeleb		Softmax	14.16
CnCeleb		AAM	13.43
CnCeleb	Self-Supervised	Contrastive	23.00
		Triplet	20.40
		ProtoLoss	18.97
		GE2E	18.76

*This line is considered as the baseline system for our experiment. Here, only labeled VoxCeleb data is used to train the embedding extractor and the trained model is directly tested on the CnCeleb evaluation set.

In this section, we first evaluate the performance of self-supervised training strategy with different contrastive objectives

and compare it with supervised training. Results are shown in Table 2. Obviously, with CnCeleb speaker labels available, the supervised training results are better than all the self-supervised training ones. Encouragingly, we find the performance self-supervised training is not that bad and the performance of the best self-supervised system is close to the Cnceleb supervised training result. Besides, the model trained on the VoxCeleb data in supervised mode performs better than the Cnceleb ones, mainly because the data amount of source domain is much larger than the target domain as shown in Table 1. Here, we consider the result of the model trained on Voxceleb in supervised training mode as the baseline for the following experiments and such result can be considered as a simple domain adaptation from source domain to target domain.

4.2. Self-Supervised Learning based Domain Adaptation.

The proposed SSDA and SSDA-Joint training strategy introduced in section 2.3 are evaluated in this section and the results are shown in Table 3.

Table 3: **Self-Supervised learning based domain adaptation results.**

Train Data	Training Mode	Target Loss	EER (%)
VoxCeleb	Supervised	-	12.11
VoxCeleb+CnCeleb	SSDA	Contrastive	20.72
		Triplet	13.66
		ProtoLoss	13.72
		GE2E	**13.34**
	SSDA-Joint	Contrastive	20.48
		Triplet	11.27
		ProtoLoss	**10.20**
		GE2E	10.24

4.2.1. *SSDA training result*

From the upper part of Table 3, we find that the performance of our proposed SSDA training strategy exceeds the simple target domain self-supervised learning by a substantial margin, which confirms that a well-pretrained model on the large scale source domain data is important. However, the SSDA training strategy still perform worse than the strong baseline system. As we assumed in section 2.3.1, simply applying self-supervised training on the target domain data may cause overfitting and the embedding extractor may lose the strong speaker discrimination ability adapted from source domain after target domain training.

4.2.2. *SSDA-Joint training result*

Plus the joint training with source domain data, SSDA-Joint training strategy further improves target domain performance compared to SSDA strategy. Notably, the SSDA-Joint training strategy with most of the contrastive learning losses performs better than the strong baseline system. Besides, the ProtoLoss performs the best in this condition and achieves 15.7% relative improvement in terms of EER compared to baseline. To the best of our knowledge, such a result exceeds the current state-of-the-art result on the CnCeleb evaluation set even without using any target domain speaker label.

4.2.3. *Evaluation on the source domain dataset*

Besides, we evaluated our best SSDA-Joint (ProtoLoss) model on the source domain evaluation set and results are shown in Table 4. Surprisingly, we find that the adaptation model trained with the SSDA-Joint training strategy performs even better than the original source domain well-trained model. The possible explanation is that joint training with target domain data may have a regularization effect on the original source domain task and much more available training data in the training process can make the embedding extractor more robust.

Table 4: **Performance of SSDA-Joint model on source domain evaluation set.** The SSDA-Joint model is evaluated on the VoxCeleb1 dataset and three official trials for VoxCeleb1 are used.

Model	EER (%)		
	Vox-O	Vox-E	Vox-H
Source Train	1.77	1.77	3.07
After Adaptation	**1.56**	**1.73**	**3.00**

4.2.4. *Comparison with fully supervised-joint training results*

To further explore the gap between the unsupervised domain adaptation results and the fully supervised joint training results, we list the relevant results in the Table 5. The supervised-joint training result can be considered as a upper-bound performance on the CnCeleb in our experiment. Notably, we find the gap between the fully supervised-joint training method and the SSDA-Joint training method is small, which means that we can save a large amount of labeling cost with minor performance degradation on the target domain by using our proposed adaptation method.

Table 5: **Comparison with supervised-joint training results**. For Supervised-Joint mode, the target domain contrastive learning loss is replaced by the supervised speaker classification loss. The source and target domain classification losses are also weighted summed as equation 10.

Train Data	Train Mode	Target Loss	EER (%)
VoxCeleb+CnCeleb	SSDA-Joint	GE2E	10.24
		ProtoLoss	10.20
	Supervised-Joint	Softmax	8.860
		AAM	9.190

5. CONCLUSION

This paper integrates the self-supervised training strategy into the domain adaptation system. Compared to the traditional UDA method, the proposed SSDA method fully leverages the potential label information from unlabeled target domain data and speaker discrimination ability from the source domain. Our proposed SSDA system achieves 10.2% EER on the CnCeleb dataset without using any target domain speaker label and even exceeds the state-of-the-art result. Besides, when source and target domain labels are both used and are jointly trained, our system makes further improvement and achieves 8.86% EER on the CnCeleb dataset.

6. ACKNOWLEDGEMENTS

This work was supported by the China NSFC projects (No. 62071288 and No. U1736202). Experiments have been carried out on the PI supercomputers at Shanghai Jiao Tong University. The author Zhengyang Chen is supported by Wu Wen Jun Honorary Doctoral Scholarship, AI Institute, Shanghai Jiao Tong University.

7. REFERENCES

[1] Yuan Liu, Yanmin Qian, Nanxin Chen, Tianfan Fu, Ya Zhang, and Kai Yu, "Deep feature for text-dependent speaker verification," *Speech Communication*, vol. 73, pp. 1–13, 2015.

[2] David Snyder, Daniel Garcia-Romero, Daniel Povey, and Sanjeev Khudanpur, "Deep neural network embeddings for text-independent speaker verification.," in *Interspeech*, 2017, pp. 999–1003.

[3] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev 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.

[4] Hossein Zeinali, Shuai Wang, Anna Silnova, Pavel Matějka, and Oldřich Plchot, "But system description to voxceleb speaker recognition challenge 2019," *arXiv preprint arXiv:1910.12592*, 2019.

[5] Koji Okabe, Takafumi Koshinaka, and Koichi Shinoda, "Attentive statistics pooling for deep speaker embedding," *arXiv preprint arXiv:1803.10963*, 2018.

[6] Yingke Zhu, Tom Ko, David Snyder, Brian Mak, and Daniel Povey, "Self-attentive speaker embeddings for text-independent speaker verification.," in *Interspeech*, 2018, pp. 3573–3577.

[7] Pooyan Safari and Javier Hernando, "Self multi-head attention for speaker recognition," *arXiv preprint arXiv:1906.09890*, 2019.

[8] Xu Xiang, Shuai Wang, Houjun Huang, Yanmin Qian, and Kai Yu, "Margin matters: Towards more discriminative deep neural network embeddings for speaker recognition," *arXiv preprint arXiv:1906.07317*, 2019.

[9] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4690–4699.

[10] Chunlei Zhang and Kazuhito Koishida, "End-to-end text-independent speaker verification with triplet loss on short utterances.," in *Interspeech*, 2017, pp. 1487–1491.

[11] Zili Huang, Shuai Wang, and Kai Yu, "Angular softmax for short-duration text-independent speaker verification.," in *Interspeech*, 2018, pp. 3623–3627.

[12] Li Wan, Quan Wang, Alan Papir, and Ignacio Lopez Moreno, "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4879–4883.

[13] Abhinav Misra and John HL Hansen, "Maximum-likelihood linear transformation for unsupervised domain adaptation in speaker verification," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 9, pp. 1549–1558, 2018.

[14] Pierre-Michel Bousquet and Mickael Rouvier, "On robustness of unsupervised domain adaptation for speaker recognition.," in *INTERSPEECH*, 2019, pp. 2958–2962.

[15] Johan Rohdin, Themos Stafylakis, Anna Silnova, Hossein Zeinali, Lukáš Burget, and Oldřich Plchot, "Speaker verification using end-to-end adversarial language adaptation," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6006–6010.

[16] Gautam Bhattacharya, Joao Monteiro, Jahangir Alam, and Patrick Kenny, "Generative adversarial speaker embedding networks for domain robust end-to-end speaker verification," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6226–6230.

[17] Wei Xia, Jing Huang, and John HL Hansen, "Cross-lingual text-independent speaker verification using unsupervised adversarial discriminative domain adaptation," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5816–5820.

[18] Zhengyang Chen, Shuai Wang, and Yanmin Qian, "Adversarial domain adaptation for speaker verification using partiallyshared network," *submitted to InterSpeech 2020*.

[19] Stephen H Shum, Douglas A Reynolds, Daniel Garcia-Romero, and Alan McCree, "Unsupervised clustering approaches for domain adaptation in speaker recognition systems," 2014.

[20] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton, "A simple framework for contrastive learning of visual representations," *arXiv preprint arXiv:2002.05709*, 2020.

[21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9729–9738.

[22] Joon Son Chung, Jaesung Huh, Seongkyu Mun, Minjae Lee, Hee Soo Heo, Soyeon Choe, Chiheon Ham, Sunghwan Jung, Bong-Jin Lee, and Icksang Han, "In defence of metric learning for speaker recognition," *arXiv preprint arXiv:2003.11982*, 2020.

[23] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3733–3742.

[24] Arsha Nagrani, Joon Son Chung, Weidi Xie, and Andrew Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Computer Speech & Language*, vol. 60, pp. 101027, 2020.

[25] Yue Fan, JW Kang, LT Li, KC Li, HL Chen, ST Cheng, PY Zhang, ZY Zhou, YQ Cai, and Dong Wang, "Cn-celeb: a challenging chinese speaker recognition dataset," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7604–7608.

[26] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society, 2011, number CONF.