INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

On the robustness of wav2vec 2.0 based speaker recognition systems

Sergey Novoselov[1], *Galina Lavrentyeva*[1], *Anastasia Avdeeva*[1], *Vladimir Volokhov*[1,2]
Nikita Khmelev[1,2], *Artem Akulov*[1,2], *Polina Leonteva*[1,2]

[1]ITMO University, St.Petersburg, [2]STC Ltd., St.Petersburg,

{novoselov, lavrentyeva, avdeeva-a, volokhov, khmelev, akulov, leonteva-p}@speechpro.com

Abstract

Recent advances in unsupervised speech representation learning discover new approaches and provide new state-of-the-art for diverse types of speech processing tasks. This paper extends the investigation of using wav2vec 2.0 deep speech representations for the speaker recognition task. It focuses on the robustness issues in different domains and considers the effectiveness of wav2vec not only on telephone and microphone speaker verification protocols but also for cross-channel task. It is concluded that powerful transformer-based speaker recognition systems can be well-generalized across variable conditions. In this study speaker recognition systems were analyzed on a wide range of well-known verification protocols. According to the results obtained in this paper we recommend to use data augmentation for fine-tuning of wav2vec based systems.

Index Terms: speaker recognition, ResNet, ECAPA-TDNN, wav2vec 2.0

1. Introduction

Today's state-of-the-art [1, 2, 3, 4] speaker recognition (SR) systems are based on very deep convolutional neural networks (*ResNets*, *ECAPA-TDNNs*) taking log Mel Filter Bank features as input and that are trained on large datasets using additive angular margin loss functions and different optimization strategies. The simple cosine or PLDA scorings are usually used as extractor back-ends.

Recent advances in unsupervised speech representation learning [5, 6, 7, 8, 9, 10] discover new approaches and provide new state-of-the-art models for diverse types of speech processing tasks including speaker recognition. The key goal of such models is speech prediction modeling [5] or speech denoising modeling [8] which can be done in an unsupervised manner. For example, papers [8, 10] report strong results for speech recognition, speaker recognition, speech separation, and speaker diarization tasks. It should be noted that an important aspect of such models is the utilization of a powerful transformer structure [7] as the backbone model which takes raw speech signals as input and incorporates a multi-head attention mechanism on the deep layers.

The authors of [11, 12] share good results of their attempts to fine-tune *wav2vec 2.0* model for VoxCeleb [13, 14] speaker recognition sets. Inspired by the success of wav2vec 2.0 in speech recognition tasks [6, 7] in our recent [15] and current works we extended the study of wav2vec 2.0 model fine-tuning for speaker recognition tasks. This paper aimed to investigate the features of these models under the varying acoustic conditions across different domains. For this purpose we used large multi-lingual wav2vec 2.0 models provided by Facebook [16] in the fairseq repository[1] as a starting point of our fine-tuning. And the experiments were conducted on a wide range of well-known benchmarks with telephone, microphone and cross-channel verification protocols. These experiments allow to consider large transformer based models in the sense of its robustness.

Taking into account our previous investigations of wav2vec systems [**?**] during the participation in the NIST SRE 2020-2021 CTS challenges [17], we used TDNN and statistic pooling layers based back-end in the wav2vec 2.0 based speaker recognition encoder network. In this paper we additionally explored the questions of optimal transformers layer selection and usefulness of audio augmentation during model fine-tuning for speaker recognition.

In addition, following our experience described in [15] we performed speaker verification for considered systems using the cl-embeddings and adaptive score normalization.

2. Speaker recognition systems

The conventional deep neural network based solution for extracting utterance-level speaker embeddings consists of three blocks: an encoder network for extracting frame-level representations from the acoustic features, pooling layer that converts variable-length frame-level features into one fixed-dimensional vector and a feed forward classification network that processes the pooling vector to produce speaker class posterior.

The role of the encoder network can be taken by a neural network of any type. We aimed to explore state-of-the-art architectures in speaker recognition and related fields for this purpose: we considered ResNet and TDNN [18] based architectures (Section 2.1) as our baseline systems. They have already shown impressive performance in the speaker verification domain. Alternative transformer-based approaches like wav2vec 2.0 model fine-tuning are described in Section 2.2.

Several papers confirm [2, 3, 18] the effectiveness of the training scheme where neural networks that are first trained on short utterances are then fine-tuned using longer utterances. We followed this approach during this study and first trained extractors on 4-6 sec speech chunks and then fine-tuned on 12-18 sec segments.

According to our experience considered deep speaker embedding extractors contain huge amounts of trainable parameters and are capable enough to solve the speaker recognition task without complex back-end or preprocessing steps. It can be trained to perform all necessary calculations by itself, given sufficient amounts of diverse training data. Following this intuition during our experiments we were mainly focused on training powerful deep speaker embedding extractors and didn't pay much attention to its front-end and back-end.

[1] https://github.com/pytorch/fairseq/tree/main/examples/wav2vec

10.21437/Interspeech.2023-881

For training and tuning processes of all our extractors the additive angular margin softmax (AAM-Softmax) based loss was used with parameters m and s set to 0.35 and 32 respectively. We used the one cycle learning rate policy [19] in all our experiments.

2.1. Baseline systems

2.1.1. Front-End processing

In this research Log Mel Filter Bank features (MFB) were used as low-level features for the baseline systems. 8kHz features were extracted from raw audio signal with 25ms frame-length and 10 ms overlap. The frequency coverage was from 20 Hz to 3700 Hz with the number of mel bins 64.

Mean Normalization (MN) over a 3-second sliding window was applied after the features were extracted. U-net-based VAD [3] was used to filter out non-speech frames.

2.1.2. ResNet101 encoder network

Proposed in 2015 for computer vision task, ResNet [20] is now one of the most popular architectures. By introducing the shortcut connections to the CNN, the ResNet model is able to build very deep neural networks and achieve remarkable performance in speaker recognition under challenging conditions [1, 21]. This network uses 2-dimensional features as input and processes them using 2-dimensional convolution in both the time and frequency domains. We used ResNet101 as a baseline model.

2.1.3. ECAPA-TDNN encoder network

Enhanced TDNN architecture with emphasized channel attention, propagation and aggregation, proposed in [22], is a modification of the standard time delay neural network (TDNN) architecture, containing squeeze-excitation (SE) blocks and Res2Net modules in the frame level with hierarchical filters for the extraction of features of different scales. To process signals of arbitrary duration, the architecture uses attentive statistic pooling instead of the conventional statistic pooling.

We used ECAPA-TDNN as our second baseline extractor. In our implementation adaptive statistic pooling and 4 SE-Res2Net Blocks with dilation values 2, 3, 4, 5 were used.

2.1.4. Scoring

We used cosine similarity to distinguish cl-embeddings extracted from the last classification linear layer [15]. Additionally, adaptive score normalization technique (adaptive s-norm) from and channel normalization postprocessing were used.

2.2. Wav2vec 2.0 based system

2.2.1. Front-End processing

Raw 16 kHz audio signal was used for our wav2vec 2.0 based extractors. Similarly to 2.1 systems U-net-based VAD [3] was used to filter out non-speech frames.

2.2.2. Wav2vec-TDNN encoder network

Wav2vec 2.0 model is a powerful transformer-based model developed for ASR tasks. It takes raw speech signals as input and incorporates a multi-head attention mechanism on the deep layers. The key aspect of training such a model is Contrastive Predicting Coding [5] self-supervised pretraining scheme. It was shown in [7] that wav2vec 2.0 model pretrained on large amounts of diverse and unlabelled data can be successfully fine-tuned to specific low resource ASR tasks.

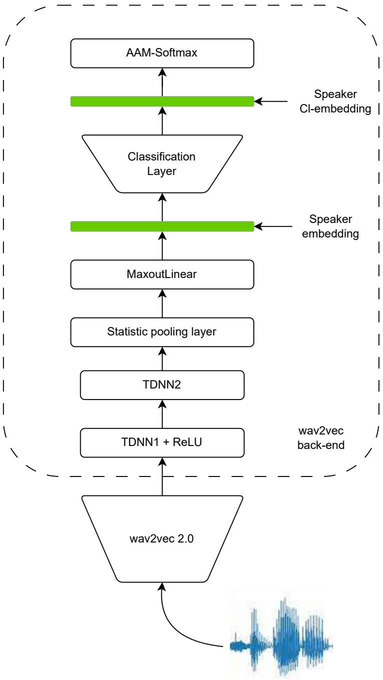

Figure 1: *Wav2vec 2.0 based speaker embeddings extractor*

As an effective wav2vec 2.0 back-end we applied two TDNN layers (the 1st with ReLU activation), statistic pooling layer to pool time series to a single vector, maxout linear layer [3, 23] to obtain speaker embedding, linear classification layer to obtain speaker cl-embedding, as presented in Figure 1. We named such a model wav2vec-TDNN. We used AAM-Softmax activation at the classification level to fine-tune the extractor. In principle, one can pass the output of the wav2vec 2.0 directly to the statistics pooling layer [11]. However, we found out that we can achieve better results if we first pass them through the sequence of TDNN layers. The role of TDNN layers is to prefilter speaker-specific information and to ”prepare” wav2vec 2.0 output time series for statistical pooling. The TDNN blocks utilize context 1 of the input features and have 2048-dimensional outputs. The obtained final speaker embedding size was 512 and the final size of the cl-embedding corresponds to the number of training classes. One additional point of interest is that wav2vec part of the extractor could be frozen while tuning for the downstream speaker recognition task. We observed that in this scenario the results can also be very impressive, but fine-tuning the whole extractor provides additional performance gains for the speaker recognition system.

Considered wav2vec-TDNN models were based on wav2vec 2.0 large. Large multi-lingual wav2vec 2.0 models like *XLSR_53*[1] and *XLS-R_1B*[2] provided by Facebook [16, 24] were used as starting points for the fine-tuning. We named the corresponding speaker embedding extractors as wav2vec-TDNN(*XLSR_53*) and wav2vec-TDNN(*XLS-R_1B*) respectively.

Scoring was similar to baseline systems (Section 2.1).

[1] https://github.com/pytorch/fairseq/tree/main/examples/wav2vec

[2] https://github.com/pytorch/fairseq/tree/main/examples/wav2vec/xlsr

3. Experimental setup

3.1. Train datasets

A wide variety of different datasets containing telephone and microphone data from proprietary datasets and from those available online were used to train the SR systems:

- Switchboard2 Phases 1, 2 and 3;
- Switchboard Cellular;
- Mixer 6 Speech;
- NIST SREs 2004–2010;
- NIST SRE 2018 (eval set);
- concatenated VoxCeleb (VC) 1 and 2;
- RusTelecom v2;
- RusIVR corpus.

RusTelecom v2 is an extended version of a private Russian corpus of telephone speech, collected by call centers in Russia. RusIVR is a private Russian corpus with telephone and media data, collected in various scenarios and recorded using different types of devices (telephone, headset, far-field microphone, etc). In total, this training set contains 532,541 records from 33,466 speakers.

3.1.1. Augmentations

For the baseline systems, we utilized standard Kaldi augmentation recipe (reverberation, babble, music and noise) with freely available MUSAN and simulated Room Impulse Response (RIR) datasets.

In the case of wav2vec 2.0 based system tuning online augmentation scheme was used for raw audio samples with the following settings:

- MUSAN additive noise with $p = 0.25$;
- RIR convolution with $p = 0.25$;
- Frequency masking with $p = 0.25$;
- Time masking with $p = 0.25$;
- Clipping distortion with $p = 0.25$.

Here p is a probability of applying augmentation type for the sample in the training batch. All considered augmentations were applied in sequence.

3.2. S-norm settings

The s-norm cohort for score normalization consisted of the following sets: NIST SRE'18 unlabeled [25], NIST SRE'16 development and unlabeled sets [26], IARPA Babel datasets [27]. We used top 200 scores to compute s-norm statistics.

3.3. Evaluation data and metrics

The following sets were used for the evaluation, data has been resampled at 16 kHz:

- **Microphone sets**: VoxCeleb1-O (VC1-O) cleaned test set [13], VOiCES development set [28];
- **Telephone sets**: NIST SRE 2018 development set [25], NIST SRE 2016 evaluation set [26], NIST SRE 2019 evaluation set [29], NIST 2020 CTS progress [17];
- **Cross channel set**: SRE 2021 challenges sets [4].

We evaluate SR systems performance in terms of Equal Error Rates (EER) and minimum detection cost functions (minDCF) with $P_{tar} = 0.01$ and $P_{tar} = 0.05$ [29].

4. Results and discussion

Tables 2 and 3 demonstrate the results of our preliminary experiments of wav2vec-TDNN(*XLSR_53*)-based systems using clean and augmented versions of VC1 train set. The results were obtained for microphone VC1-O (cleaned) and telephone SRE'18 dev evaluation protocols. The performance of the system depending on wav2vec 2.0 transformer layer selection is shown on the Tables. It can be seen from the results of Table 3 that there is no need to use the entire deep wav2vec 2.0 encoder architecture with 24 layers for such SR task. The 6th encoder layer provides speaker recognition results comparable to other deeper transformer encoder layers. Thus in our further experiments with wav2vec-TDNN(*XLSR_53*)-based systems we used 6th layer transformer encoder network before TDNN block. For the wav2vec-TDNN based on *XLS-R_1B* architecture, 12th layer was chosen as optimal. The online augmentations did not help to improve SR systems performance in the case of a small VC1 tuning set and 20 epochs tuning procedure. We guess the reason for that is a lack of clean in-domain data which the network "has seen" during the training in this scenario.

Table 4 reveals the positive effect of using augmentation procedure during wav2vec-TDNN(*XLSR_53*) extractor fine-tuning for SR. One can see that data augmentation improves system robustness in telephone domain when only microphone data (VC1 and VC2) is used for training. Moreover, Tables 2, 3 and 4 show impressive and state-of-the-art results of the systems fine-tuned on relatively small sets VC1 (1211 speakers) and VC1+VC2 (7205 speakers).

For comparison of baseline and new wav2vec 2.0 based deep speaker embedding extractors performance on different evaluation protocols see Table 1. These results show the robustness of new encoders to different acoustic conditions in comparison to considered baseline systems. Our best and largest wav2vec-TDNN(*XLS-R_1B*) model demonstrates strong stability across microphone and telephone evaluation data achieving $EER = 0.69\%$ on VC1-O (cleaned) and $EER = 1.71\%$ on SRE'19 eval protocols. One should note that there is a difference in baseline and wav2vec 2.0 models complexity in terms of the number of trainable parameters. According to our results increasing the model complexity for ResNet or ECAPA-TDNN did not lead to better robustness to different domains. We also tried to add SpecAugment for baseline systems training but did not observe any performance improvements.

Another thing we should note is that our attempts to train wav2vec-TDNN SR systems from scratch were unsuccessful. Thus we conclude that an unsupervised autoregressive pretraining scheme (for example with Contrastive Predictive Coding loss) efficiently utilizes the power of unlabeled data and opens the door to powerful transformer-based speaker embedding extractors.

5. Conclusions

Large transformer-based speaker embedding extractors can be developed with the help of unsupervised speech representation learning schemes. Our experiments for wav2vec 2.0 on a wide range of verification protocols reveal that such models are powerful and robust across different acoustic conditions including cross-channel. Presented wav2vec-TDNN models fine-tuned on diverse training sets with augmentations demonstrate good robustness and generalization across different acoustic domains.

It was shown that fine-tuning of wav2vec-TDNN architectures for specific domains can be done on relatively small sets of data. Using data augmentation during fine-tuning provides additional performance gains in speaker verification.

Table 1: *Speaker recognition evaluations on different test protocols for baseline systems and proposed wav2vec 2.0 based systems in terms of EER[%] / minDCF(0.05)*

System	#Params, M	Test datasets						
		SRE'18 dev	SRE'16 eval	SRE'19 eval	VC1-O (cleaned)	VOiCES dev	CTS'20 progress[3]	SRE'21 eval[4]
				Baseline encoders				
ResNet101	27.5	3.28/0.118	5.01/0.237	2.39/0.134	1.78/0.105	1.81/0.110	2.75/0.097	5.41/0.344
ECAPA-TDNN	29	4.14/0.152	8.59/0.337	2.97/0.165	1.87/0.123	2.02/0.123	2.91/0.109	6.26/0.398
ResNet101 + ECAPA TDNN	56.5	3.17/0.114	4.87/0.221	2.12/0/122	1.35/0.086	1.31/0.081	2.71/0.085	4.74/0.299
				New encoders				
wav2vec-TDNN (XLSR_53)	98	3.07/0.137	4.18/0.206	2.34/0.142	0.82/0.052	0.99/0.06	**2.25/0.080**	4.43/0.283
wav2vec-TDNN (XLS-R_1B)	265	**2.94/0.083**	**3.13/0.161**	**1.71/0.097**	**0.69/0.040**	1.02/0.057	3.61/0.080	**3.59/0.281**

Table 2: *Results of speaker verification on VC1-O (cleaned) and SRE'18 dev sets in dependence of wav2vec 2.0 encoder output layer selection for wav2vec-TDNN(XLSR_53)*[1]

Layer	Train set	VC1-O (cleaned)		SRE'18 dev	
		EER	DCF(0.01)	EER	DCF(0.01)
3	VC1	2.54	0.29	13.58	0.62
6		1.82	0.22	10.19	0.51
9		1.76	0.197	10.5	0.48
12		1.71	0.21	10.58	0.52
18		**1.61**	**0.17**	**9.97**	**0.44**
24		na[2]	na	na	na

Table 3: *Results of speaker verification on VC1-O (cleaned) test and SRE'18 dev sets in dependence of wav2vec 2.0 encoder output layer selection for wav2vec-TDNN(XLSR_53)*[1]

Layer	Train set	VC1-O (cleaned)		SRE'18 dev	
		EER	DCF(0.01)	EER	DCF(0.01)
3	VC1 + augs	3.47	0.327	12.22	0.55
6		2.37	**0.227**	**9.78**	**0.45**
9		2.23	0.267	10.88	0.48
12		2.38	0.321	10.34	0.45
18		**2.21**	0.243	11.06	0.54
24		16.62	0.99	30	1

Table 4: *Results of speaker verification on VC1-O (cleaned) test and SRE'18 development sets for wav2vec-TDNN(XLSR_53)*[1]

Train set	VC1-O (cleaned)		SRE'18 dev	
	EER	DCF(0.01)	EER	DCF(0.01)
VC1+VC2	0.86	0.082	9.07	0.47
VC1+VC2 + augs	**0.84**	**0.058**	**7.5**	**0.38**

6. Acknowledgements

This work was supported by the Analytical Center for the Government of the Russian Federation (IGK 000000D730321P5Q0002), agreement No. 70-2021-00141.

We express our gratitude and deep appreciation to our colleagues from Speaker and Speech Recognition Teams.

7. References

[1] Hossein Zeinali, Shuai Wang, Anna Silnova, Pavel Matějka, and Oldřich Plchot, "BUT system description to VoxCeleb speaker recognition challenge 2019," Tech. Rep., 2019.

[2] Daniel Garcia-Romero, Greg Sell, and Alan Mccree, "MagNetO: X-vector magnitude estimation network plus offset for improved speaker recognition," in *Proc. Odyssey 2020 the speaker and language recognition workshop*, 2020.

[3] Aleksei Gusev, Vladimir Volokhov, Tseren Andzhukaev, Sergey Novoselov, Galina Lavrentyeva, Marina Volkova, Alice Gazizullina, Andrey Shulipa, Artem Gorlanov, Anastasia Avdeeva, Artem Ivanov, Alexander Kozlov, Timur Pekhovsky, and Yuri Matveev, "Deep speaker embeddings for far-field speaker recognition on short utterances," *arXiv preprint arXiv:2002.06033*, 2020.

[4] Seyed Omid Sadjadi, Craig Greenberg, Elliot Singer, Lisa Mason, and Douglas Reynolds, "The NIST 2021 speaker recognition evaluation plan," 2021.

[5] Aaron van den Oord, Yazhe Li, and Oriol Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[6] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli, "wav2vec: Unsupervised pre-training for speech recognition," 2019.

[7] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," 2020.

[8] Sanyuan Chen, Chengyi Wang, et al., "WavLM: Large-scale self-supervised pre-training for full stack speech processing," *arXiv preprint arXiv:2110.13900*, 2021.

[9] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed, "HuBERT: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021.

[10] Sanyuan Chen, Yu Wu, Chengyi Wang, Zhengyang Chen, Zhuo Chen, Shujie Liu, Jian Wu, Yao Qian, Furu Wei, Jinyu Li, and Xiangzhan Yu, "UniSpeech-SAT: Universal speech representation learning with speaker aware pre-training," *CoRR*, vol. abs/2110.05752, 2021.

[11] Zhiyun Fan, Meng Li, et al., "Exploring wav2vec 2.0 on speaker verification and language identification," *arXiv preprint arXiv:2012.06185*, 2020.

[12] Nik Vaessen and David A van Leeuwen, "Fine-tuning wav2vec2 for speaker recognition," *arXiv preprint arXiv:2109.15053*, 2021.

[13] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman, "VoxCeleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[1] Model tuned during 20 epochs

[2] No convergence achieved

[3] evaluated using NIST SRE platform

[4] evaluated using NIST SRE 2021 scoring tool

[14] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman, "VoxCeleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[15] Galina Lavrentyeva, Sergey Novoselov, Vladimir Volokhov, Anastasia Avdeeva, Aleksei Gusev, Alisa Vinogradova, Igor Korsunov, Alexander Kozlov, Timur Pekhovsky, Andrey Shulipa, Evgeny Smirnov, and Vasily Galyuk, "STC Speaker Recognition System for the NIST SRE 2021," in *Proc. The Speaker and Language Recognition Workshop (Odyssey 2022)*, 2022, pp. 354–361.

[16] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli, "fairseq: A fast, extensible toolkit for sequence modeling," *arXiv preprint arXiv:1904.01038*, 2019.

[17] Seyed, Craig Greenberg, Elliot Singer, Douglas Olson, and Lisa Mason, "NIST 2020 CTS speaker recognition challenge evaluation plan," 2020.

[18] Daniel Garcia-Romero, David Snyder, Gregory Sell, Alan McCree, Daniel Povey, and Sanjeev Khudanpur, "X-vector DNN refinement with full-length recordings for speaker recognition," in *Interspeech*, 2019, pp. 1493–1496.

[19] Leslie N. Smith and Nicholay Topin, "Super-convergence: Very fast training of neural networks using large learning rates," 2018.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015.

[21] Jesús Antonio Villalba López, Daniel Garcia-Romero, Nanxin Chen, Gregory Sell, Jonas Borgstrom, Alan McCree, Leibny Paola García-Perera, Saurabh Kataria, Phani Sankar Nidadavolu, Pedro Torres-Carrasquiilo, and Najim Dehak, "Advances in speaker recognition for telephone and audio-visual data: the JHU-MIT submission for NIST SRE19," 2020.

[22] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck, "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification," *arXiv preprint arXiv:2005.07143*, 2020.

[23] Sergey Novoselov, Andrey Shulipa, Ivan Kremnev, Alexandr Kozlov, and Vadim Shchemelinin, "On deep speaker embeddings for text-independent speaker recognition," *arXiv preprint arXiv:1804.10080*, 2018.

[24] Arun Babu, Changhan Wang, Andros Tjandra, Kushal Lakhotia, Qiantong Xu, Naman Goyal, Kritika Singh, Patrick von Platen, Yatharth Saraf, Juan Pino, Alexei Baevski, Alexis Conneau, and Michael Auli, "XLS-R: Self-supervised cross-lingual speech representation learning at scale," *arXiv preprint arXiv:2111.09296*, 2021.

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

[26] Seyed Sadjadi, Timothée Kheyrkhah, Audrey Tong, Craig Greenberg, Douglas Reynolds, Elliot Singer, Lisa Mason, and Jaime Hernandez-Cordero, "The 2016 NIST speaker recognition evaluation," in *Interspeech*, 2017, pp. 1353–1357.

[27] Christopher Cieri, Mark Liberman, et al., "From 'solved problems' to new challenges: A report on LDC activities," in *Proceedings of the LREC 2018*, 2018.

[28] Mahesh Kumar Nandwana, Julien van Hout, Mitchell McLaren, Colleen Richey, Aaron Lawson, and Maria Alejandra Barrios, "The VOiCES from a distance challenge 2019 evaluation plan," *arXiv preprint arXiv:1902.10828*, 2019.

[29] Seyed Omid Sadjadi, Craig Greenberg, et al., "The 2019 NIST speaker recognition evaluation CTS challenge," in *Speaker Odyssey*, 2020, vol. 2020, pp. 266–272.