LARGE-SCALE SELF-SUPERVISED SPEECH REPRESENTATION LEARNING FOR AUTOMATIC SPEAKER VERIFICATION

Zhengyang Chen[1,2,] , Sanyuan Chen[2], Yu Wu[2], Yao Qian[2], Chengyi Wang[2]*
Shujie Liu[2], Yanmin Qian[1], Michael Zeng[2]

[1] MoE Key Lab of Artificial Intelligence, AI Institute ,
X-LANCE Lab, Department of Computer Science and Engineering, Shanghai Jiao Tong University
[2]Microsoft Corporation

ABSTRACT

The speech representations learned from large-scale unlabeled data have shown better generalizability than those from supervised learning and thus attract a lot of interest to be applied for various downstream tasks. In this paper, we explore the limits of speech representations learned by different self-supervised objectives and datasets for automatic speaker verification (ASV), especially with a well-recognized SOTA ASV model, ECAPA-TDNN [1], as a downstream model. The representations from all hidden layers of the pre-trained model are firstly averaged with learnable weights and then fed into the ECAPA-TDNN as input features. The experimental results on Voxceleb dataset show that the weighted average representation is significantly superior to FBank, a conventional handcrafted feature for ASV. Our best single system achieves 0.537%, 0.569%, and 1.180% equal error rate (EER) on the three official trials of VoxCeleb1, separately. Accordingly, the ensemble system with three pre-trained models can further improve the EER to 0.479%, 0.536% and 1.023%. Among the three evaluation trials, our best system outperforms the winner system [2] of the VoxCeleb Speaker Recognition Challenge 2021 (VoxSRC2021) on the VoxCeleb1-E trial.

Index Terms— representation learning, self-supervised pre-train, speaker verification

1. INTRODUCTION

Recent years have witnessed significant improvements in automatic speaker verification (ASV) tasks. Researchers have developed various neural network architectures [1, 3, 4, 5], training objectives [6, 7, 8, 9], pooling functions [10, 11] to push the limits of the system performance. However, these techniques always require large-amount well-labeled data. It is a challenge to collect large-scale labeled data for real applications due to the privacy issue of speaker information. Over the past years, pre-trained models have become the de-facto standard for state-of-the-art performance on many natural language processing (NLP) tasks. Inspired by the great success of BERT [12] and GPT [13], a series of work in the speech community, e.g. wav2vec 2.0 [14] and HuBERT [15], have been proposed to leverage large-scale unlabeled data, showing the impressive results on the automatic speech recognition (ASR) tasks.

For the speaker verification field, many researchers have designed specific losses to train the extractor of speaker embeddings from the unlabeled data under an assumption that there is only one speaker in one utterance [16, 17, 18]. Such an assumption may limit

---

the application for un-supervised speaker verification training on the unlimited data from the internet. The Wav2Vec 2.0 [14] and Hu-BERT [15] rely less on such assumption. These two pre-trained models have shown that they can capture phonetic structure information contained in speech and thus benefit ASR. It is an interesting research topic to probe the nature of the representations learned by different layers of pre-trained models [19, 20]. The effectiveness of Wav2vec 2.0 in a two-stage training process of pre-trained and fine-tuning has been demonstrated on both speaker verification and language recognition tasks in [21]. Besides, [22] introduces a benchmark to evaluate the performance of pre-trained models and shows the better performance of the speech representations learned from large-scale unlabeled data, by comparing with Fbank, on various downstream tasks including ASV. In order to minimize architecture changes and fine-tuning to solve all downstream tasks, the works above only use a simple downstream model and train the system on a small speaker verification dataset Voxceleb1 [23] for ASV task. However, whether the speech representations can also benefit the state-of-the-art (SOTA) ASV systems is still an open question.

In this paper, the speech representations learned from large-scale unlabeled data are extensively investigated on a benchmark dataset for speaker verification. The major contribution of this paper is four-fold as follows:

1. To the best of our knowledge, it is the first attempt to use the speech representation learned from large-scale unlabeled data to improve the performance of the SOTA speaker verification model (i.e., ECAPA-TDNN [1]) on Voxceleb dataset.

2. Instead of using the representations only from the final layer of the pre-trained model, we employ a weighted average of the representations from all hidden layers to fully leverage the speaker-related information embedded in the whole model.

3. We conduct a comprehensive study on the performance of pre-trained models with different learning methods, model sizes and large-scale training datasets.

4. A detailed analysis based on learnable weights is performed for probing layer-wise speaker information embedded in the pre-trained models.

2. RELATED WORK

Speech signals contain all kinds of information, such as phonetic structure, emotion, speaker indentity, etc. The Fbank and MFCC are the most commonly used handcrafted acoustic features, which demonstrate sound characteristics in the frequency domain. In addition, researchers have been doing lots of feature engineering to

improve their performance, e.g., delta features to capture temporal dynamics of Fbank or MFCC. The authors in [24] combined the articulation rate filter with the constant Q cepstral coefficients (CQCCs) [25] in the speaker verification task and achieved significant improvement compared to MFCC baseline. In order to make better use of the powerful learning ability of neural networks, Mirco et al. [26] and Jee-weon et al. [27] have tried to used convolution neural network to learn task-specific features from raw audio signals and achieved comparable performance with handcrafted feature.

Recently, speech representation learning by leveraging unlabeled data is gradually emerging. It is commonly believed that the pre-trained models by self-supervised learning have a good generalizability and a simple classifier added on the top of the representations from these pre-trained models can obtain decent performance for many downstream tasks, even with a limited amount of labeled data. Self-supervised learning for speech representations can be categorized into three approaches: 1) reconstruction learning aims to reconstruct the original input using information extracted from past time steps or masked inputs; 2) Contrastive learning learns high-level representations by solving a contrastive task in the latent embedding space; 3) multi-task learning with multiple objectives and multiple inputs. A review of these approaches is given in [22].
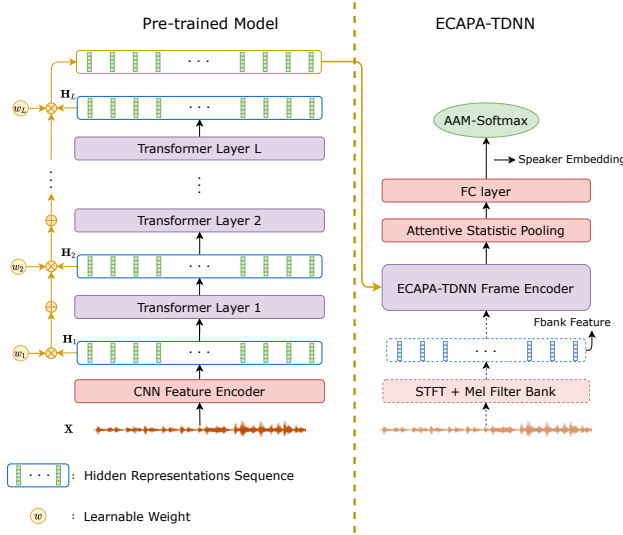

Fig. 1. **Leverage Representations from Pre-trained Model**

3. METHODS

3.1. Pre-train for Representation Learning

In this study, we leverage the representations from Wav2Vec 2.0 [14], HuBERT [15] and UniSpeech-SAT [28] to do speaker verification task. These three models use different methods to learn the feature representation. The Wav2Vec 2.0 model uses a contrastive loss to distinguish a true speech segment from negatives. The goal of HuBERT is to predict the weak supervised label for the mask frames. UniSpeech-SAT integrates an utterance-wise contrastive loss into Hubert-like representation learning that forces speaker-related information into the learned representation. Despite the different training objectives for the pre-trained models described above, they share the similar model structures. As shown in the left part of Figure 1, these three pre-trained models all consist of a convolutional feature extractor and a deep transformer [29] network as the encoder. Math-

ematically, given an input wavform $\mathbf{X} = \{x_1 \dots x_N\}_{t=1}^N$ where N is the number of sampling points, the CNN feature encoder convolves the sample points to a sequence of feature vector, $\mathbf{H}_0 = \{\mathbf{h}_{1,0} \dots \mathbf{h}_{T,0}\}_{t=1}^T$. Then the sequence of feature vector is fed to the Transformer model, yielding a hidden state for each frame at the l-th layer $\mathbf{H}_l = \{\mathbf{h}_{1,l} \dots \mathbf{h}_{T,l}\}_{t=1}^T$, where $l \in \{1, \dots, L\}$.

3.2. Leverage Representations from pre-trained Model

3.2.1. Downstream Speaker Verification Model

In [21], the authors added an average pooling layer and a fully connected layer with a task-specific loss on the top of pre-trained models and achieved comparable results with the systems using handcrafted features. In [22], x-vector [4] is used as the downstream model. To push the limit of the performance of the downstream task, we use the state-of-the-art speaker verification system ECAPA-TDNN [1] as the downstream model. Compared to x-vector, ECAPA-TDNN has a more advanced design, e.g. Squeeze-Excitation Res2Blocks [30, 31] and multi-layer feature aggregation, which significantly improves system performance. The brief structural framework of ECAPA-TDNN is shown as the right part of Figure 1. The model takes the sequence of the Fbank feature as input. Then, the frame encoder extracts speaker information from each input frame and the statistic pooling layer transforms the variable length input sequence to fix-dimensional representation. Finally, a fully connected (FC) layer is added to extract speaker embedding. To leverage the representations learned from the pre-trained models, we can replace Fbank with the last-layer outputs of pre-trained models and feed them into the ECAPA-TDNN.

3.2.2. Explore Speaker Information in Pre-trained Model

The pre-trained model, which has seen tons of audio data, should have good generalization for various downstream tasks. However, the results in [21] didn't show the superiority of the pre-trained representation compared to handcrafted feature. The objectives of the most pre-trained tasks are not directly related to speaker recognition. The layers close to the final objectives will contain more information related to the training loss. It could be better to discover the speaker information from the low layers of the pre-trained model.

Here, similar to the implementation in [22, 32], we introduce a learnable weight, w_l, for hidden states $\mathbf{H}_l, l \in \{0, \dots, L\}$ from each layer in pre-trained model. Rather feeding the outputs from the last layer of the pre-trained model, i.e. $\mathbf{H}_L$, to the downstream model, we weighted average the hidden states of each layer to generate the frame representation $\mathbf{o}_t = \sum_{l=0}^L w_l \cdot h_{l,t}$. Then, we replace the Fbank feature fed into the ECAPA-TDNN with the weighted average representations to extract speaker embedding $\mathbf{e}$:

$$\mathbf{e} = \text{ECAPA-TDNN}(\mathbf{o}_1 \dots \mathbf{o}_T) \qquad (1)$$

Same as the implementation in [1], we also use the additive angular margin (AAM) [33] loss in the training process for model optimization.

The training pipeline is mainly divided into two stages. In the first stage, the pre-trained model is fixed. We only update the ECAPA-TDNN and the weight w for all the hidden states. Then, we fine-tune all the parameters for pre-trained model and ECAPA-TDNN.

4. EXPERIMENTAL SETUP

To analyze the effectiveness of pre-trained model representation for speaker verification task, we trained and evaluated the downstream speaker verification model using Voxceleb1 [23] and Voxceleb2 [34] datasets. All three official trial lists Vox1-O, Vox1-E and Vox1-H are

Table 1. **The detailed information of pre-trained models used in our experiments and down-stream task model.** For the UniSpeech-SAT_* models, we use Librivox (60k hrs), VoxPopuli (24k hrs) and Gigaspeech (10k hrs, English) to form the 90k hours training data. The layer # column only counts the transformer layer in the pre-training model.

Pre-training/Down-stream Model	Layer #	Parameter #	Training Data		
			Duration	Sources	Language
HuBERT_Base	12	∼95M	960 hrs	Librispeech	English
HuBERT_Large	24	∼316M	60k hrs	Librivox	English
Wav2Vec2.0_Large (XLSR)	24	∼316M	56k hrs	Multilingual LibriSpeech, CommonVoice, BABLE	Over 36 languages
UniSpeech-SAT_Base	12	∼95M	94k hrs	Librivox, VoxPopuli, Gigaspeech	English
UniSpeech-SAT_Large	24	∼316M	94k hrs	Librivox, VoxPopuli, Gigaspeech	English
ECAPA-TDNN (small) [1]	-	∼6M	2.36k hrs	Voxceleb2 (Youtube)	Multi-Lingual (mostly English)

used to evaluate the system performance. When implementing our baseline models using the handcrafted acoustic feature, we extract 40-dimensional Fbank feature with 25ms window size and 10ms frame shift. We didn't do voice activity detection (VAD) processing for the Voxceleb data. Besides, we also did data augmentation for the training data using the MUSAN [35] noise and RIR [1] reverberation with probability 0.6 in online mode.

The detailed information about the pre-trained models used in our experiments and the speaker verification downstream models is listed in Table 1. The HuBERT_Base, HuBERT_Large and Wav2vec2.0_Large (XLSR) models are released by Fairseq sequence modeling toolkit [2]. The results in [22] show that the Wav2vec2.0_Base performed worse than HuBERT_Base on speaker-related task and we didn't use it here. UniSpeech-SAT is a model proposed recently, which explicitly models the speaker information in pre-training process. It introduces utterance contrastive loss to model the single speaker information, where the positive instances are hidden states in the same utterance while the negative instances are hidden states in other utterances. Moreover, UniSpeech-SAT uses more synthesis or public available data compared to HuBERT. For downsteam task model, we use the small ECAPA-TDNN in [1].

We trained all the models with Additive Angular Margin Loss (AAM) [33] and set the margin to 0.2. During the training process, we randomly sampled 3s segment from each utterance to construct training batch. For the two-stage training pipeline described in section 3.2.2, we first fixed the pre-trained model and trained for 10 epochs. Then, we fine-tuned all the parameters for another 5 epochs. Besides, to further improve our best system, we did large margin fine-tuning [36] by randomly sampling 6s segments and set the AAM margin to 0.5 to train extra 2 epochs.

During the evaluation, we use the cosine score to measure the similarity for trial pairs. We also use the adaptive s-norm [37, 38] to normalize the scores in our experiment. The embeddings extracted from the training set are averaged according to the speaker label and used as the imposter cohort. We set the imposter cohort size to 600 in our experiment. When doing quality-aware score calibration [36], we randomly generated 30k trials based on the voxceleb2 test set to train our calibration model.

5. EVALUATION RESULTS

5.1. Comparison with Handcrafted Acoustic Feature

First, we will compare the speech representations extracted from pre-trained models with the commonly used handcrafted feature. The experiments in [21] have shown that Wav2Vec 2.0 pre-trained models contain speaker information and can achieve comparable performance with the handcrafted acoustic feature. Different from [21], in

Table 2. **Comparison with traditional acoustic feature based on Voxceleb1.** Here, we trained all the models on Voxceleb1 dev set and evaluated on Vox1-O trial. We fixed pre-trained model in the training process and only use them to extract speech representation.

Feature	Aug	Pretrain Feature	Vox1-O EER (%)
Fbank	✗	-	3.899
HuBERT_Base	✗	Last	3.691
HuBERT_Base	✗	Hidden	2.117
Fbank	✓	-	2.371
HuBERT_Base	✓	Last	3.079
HuBERT_Base	✓	Hidden	1.861
UniSpeech-SAT_Base	✓	Hidden	1.632

our experiments, we directly replaced the handcrafted feature fed to the speaker verification model ECAPA-TDNN with the representations from pre-trained models. Besides, we explored to leverage the representations from pre-trained models in two different ways, using the representation from the last layer or weighted averaging all the hidden representations. The results are shown in Table 2. From the upper part of the table, we find that the last layer representation and all hidden layers' representation from the pre-trained model both perform better than the handcrafted feature Fbank. Encouragingly, the performance of weighted averaging hidden representation exceeds the Fbank by a very large margin (46% relatively). Then, we augment the training data and the results are listed in the bottom part of Table 2. With data augmentation, all the results are further improved and the weighted averaging hidden representations also shows superiority over the Fbank feature. For the experiments in the following sections, we will use the weighted average hidden representations for pre-trained model and augment the training data.

5.2. Comparison among Different Pre-trained Models

To further improve the effectiveness of the representations from pre-trained models, we trained the model on a larger dataset, Voxceleb2_dev, and compared different pre-trained models and training strategies. All the results are shown in Table 3. The results show that all the large models perform better than Fbank feature on both Vox1_dev and Vox2_dev setup. When we unfix the pre-trained model and jointly fine-tune the pre-trained model and downstream model, further improvements can be achieved. The improvement from pre-trained model fine-tuning is more obvious on Vox2_dev setup than Vox1_dev setup. Besides, the Wav2vec2.0_Large (XLSR) and UniSpeech-SAT_Large pre-trained models perform better than the HuBERT_Large after fine-tuning. As shown in table 1, the training set size of the Wav2vec2.0_Large (XLSR) and HuBERT_Large is comparable. However, the training data for Wav2vec2.0_Large (XLSR) is more diverse and more matched with Voxceleb data, enabling it to be more suitable for this downstream task. Moreover, the UniSpeech-SAT_Large model with more training data performs the best among most of the trials. Compared to Fbank feature, represen-

---

[1] https://www.openslr.org/28/
[2] https://github.com/pytorch/fairseq

Table 3. **Results with different pre-trained models and different training strategies.** Here, we did data augmentation for all the experiments. In the last Ensemble line, we weighted average the scores of our best three systems after score calibration. The weight in the score average is decided according to the performance of the single system. Besides, the Large model performs much better than the Base model and we only list a part of the results for the Base model because of the space limit.

Train Data	Large Margin Finetune	Score Calibration	Fix Pretrain	Feature	EER (%)		
					Vox1-O	Vox1-E	Vox1-H
Vox1_dev	✗	✗	-	Fbank	2.371	-	-
			✓	UniSpeech-SAT_Base	1.632	-	-
			✓	HuBERT_Large	1.436	-	-
			✓	Wav2Vec2.0_Large (XLSR)	1.362	-	-
			✓	UniSpeech-SAT_Large	1.249	-	-
			✗	UniSpeech-SAT_Base	1.611	-	-
			✗	HuBERT_Large	1.404	-	-
			✗	Wav2Vec2.0_Large (XLSR)	1.335	-	-
			✗	UniSpeech-SAT_Large	1.218	-	-
Vox2_dev	✗	✗	-	Fbank (ECAPA-TDNN small [1])	1.010	1.240	2.320
	✗	✗	-	Fbank (ECAPA-TDNN large [1])	0.870	1.120	2.120
	✗	✗	-	Fbank	1.080	1.200	2.127
	✗	✗	✓	UniSpeech-SAT_Base	1.095	1.152	2.221
	✗	✗	✓	HuBERT_Large	0.914	0.948	1.759
	✗	✗	✓	Wav2Vec2.0_Large (XLSR)	1.021	0.962	1.782
	✗	✗	✓	UniSpeech-SAT_Large	0.750	0.813	1.649
	✗	✗	✗	UniSpeech-SAT_Base	1.005	0.933	1.866
	✗	✗	✗	HuBERT_Large	0.814	0.777	1.505
	✗	✗	✗	Wav2Vec2.0_Large (XLSR)	0.803	0.729	1.394
	✗	✗	✗	UniSpeech-SAT_Large	0.696	0.685	1.433
	✓	✗	✗	HuBERT_Large	0.723	0.706	1.317
	✓	✗	✗	Wav2Vec2.0_Large (XLSR)	0.734	0.677	1.235
	✓	✗	✗	UniSpeech-SAT_Large	0.633	0.625	1.294
	✓	✓	✗	HuBERT_Large	0.590	0.654	1.227
	✓	✓	✗	Wav2Vec2.0_Large (XLSR)	0.585	0.625	**1.138**
	✓	✓	✗	UniSpeech-SAT_Large	**0.537**	**0.569**	1.180
	✓	✓	✗	Ensemble	0.479	0.536	1.023

tations from this model achieved ∼30% relative EER improvement on all three trials for the Voxceleb1 evaluation set.

In [36], the authors introduced a large margin fine-tuning strategy and quality-aware score calibration to the speaker verification task and achieved impressive improvement. Here, we also leverage these two strategies in our experiments to push the performance limit. The corresponding results are listed at the bottom part in Table 3. With these two strategies, our best system exceeds the state-of-the-art system [2] (Vox1-O: 0.461, Vox1-E: 0.634, Vox1-H: 0.993) in VoxSRC challenge 2021 on Vox-E trial.
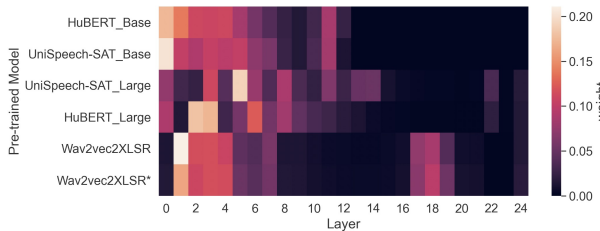

Fig. 2. **The visualization of the normalized weight values in the proposed architecture show as Figure 1**. The output from the layer 0 corresponds to the transformer input. * in the figure means that the pre-trained model is unfixed during downsteam task training. It should be noted that base models only have 12 layers and other models have 24 layers.

5.3. Analysis Speaker Information in Pre-trained Model

The results in Section 5 have shown that it is better to leverage the representations from all the hidden layers rather than the last layer. Thus, it could be necessary and meaningful to explore which layer contains more speaker information than the others. We visualize the normalized weight value for all the layers' output in Figure 2. The figure shows that the speaker information at the lower layers of pre-trained models is more discriminative than those at the higher layers for ASV task . This phenomenon is reasonable because the training objectives for the pre-trained models used in our experiments are more related to the speech recognition task. For large pre-trained models in our experiments, i.e. UniSpeech-SAT_Large, HuBERT_Large and Wav2vec2.0_Large (XLSR), the learned weights assigned to the higher layers are much smaller than those of lower layers, which indicates that we might be able to directly throw away these higher layers to reduce model size.

6. CONCLUSION

In this paper, we leverage the representations extracted from pre-trained models trained on large-scale unlabeled data in speaker verification task. In our experiments, we first compared such representations with handcrafted Fbank feature and verify the superiority of pre-trained representations. To comprehensively explore speaker information in the pre-trained model, we make the model learn the weights automatically for all the hidden states of the pre-trained model and achieve significant performance improvement compared to the baseline. By visualizing the learned weights, we find the lower layers of the pre-trained model can capture more speaker-related information than those of higher layers. Despite the significant improvement benefiting from the pre-trained model, there is still a relatively small performance gap (on two evaluation sets) between our system and the best system [2] in the VoxSRC2021 challenge, which has a more aggressive augmentation strategy and dedicated training objectives. In the future, we will incorporate the better training setup in [2] for our system to further push the limit of speaker verification performance.

7. REFERENCES

[1] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification," in *Proc. Interspeech 2020*, 2020, pp. 3830–3834.

[2] M. Zhao, Y. Ma, M. Liu, and M. Xu, "The speakin system for voxceleb speaker recognition challange 2021," *arXiv preprint arXiv:2109.01989*, 2021.

[3] Y. Liu, Y. Qian, N. Chen, T. Fu, Y. Zhang, and K. Yu, "Deep feature for text-dependent speaker verification," *Speech Communication*, vol. 73, pp. 1–13, 2015.

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

[5] H. Zeinali, S. Wang, A. Silnova, P. Matějka, and O. Plchot, "But system description to voxceleb speaker recognition challenge 2019," *arXiv preprint arXiv:1910.12592*, 2019.

[6] X. Xiang, S. Wang, H. Huang, Y. Qian, and K. Yu, "Margin matters: Towards more discriminative deep neural network embeddings for speaker recognition," in *Proc. APSIPA ASC 2019*. IEEE, 2019, pp. 1652–1656.

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

[8] Z. Huang, S. Wang, and K. Yu, "Angular softmax for short-duration text-independent speaker verification." in *Interspeech*, 2018, pp. 3623–3627.

[9] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *Proc. IEEE ICASSP 2018*. IEEE, 2018, pp. 4879–4883.

[10] K. Okabe, T. Koshinaka, and K. Shinoda, "Attentive statistics pooling for deep speaker embedding," *arXiv preprint arXiv:1803.10963*, 2018.

[11] Y. Zhu, T. Ko, D. Snyder, B. Mak, and D. Povey, "Self-attentive speaker embeddings for text-independent speaker verification." in *Interspeech*, vol. 2018, 2018, pp. 3573–3577.

[12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[13] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," 2018.

[14] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *arXiv preprint arXiv:2006.11477*, 2020.

[15] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *arXiv preprint arXiv:2106.07447*, 2021.

[16] H. Zhang, Y. Zou, and H. Wang, "Contrastive self-supervised learning for text-independent speaker verification," in *Proc. IEEE ICASSP 2021*. IEEE, 2021, pp. 6713–6717.

[17] W. Xia, C. Zhang, C. Weng, M. Yu, and D. Yu, "Self-supervised text-independent speaker verification using prototypical momentum contrastive learning," in *Proc. IEEE ICASSP 2021*. IEEE, 2021, pp. 6723–6727.

[18] D. Cai, W. Wang, and M. Li, "An iterative framework for self-supervised deep speaker representation learning," in *Proc. IEEE ICASSP 2021*. IEEE, 2021, pp. 6728–6732.

[19] G. Jawahar, B. Sagot, and D. Seddah, "What does BERT learn about the structure of language?" in *Proc. ACL*, Jul. 2019, pp. 3651–3657.

[20] A. Pasad, J. Chou, and K. Livescu, "Layer-wise analysis of a self-supervised speech representation model," *CoRR*, vol. abs/2107.04734, 2021.

[21] Z. Fan, M. Li, S. Zhou, and B. Xu, "Exploring wav2vec 2.0 on Speaker Verification and Language Identification," in *Proc. Interspeech 2021*, 2021, pp. 1509–1513.

[22] S.-w. Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin *et al.*, "Superb: Speech processing universal performance benchmark," *arXiv preprint arXiv:2105.01051*, 2021.

[23] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[24] M. Todisco, H. Delgado, and N. W. Evans, "Articulation rate filtering of cqcc features for automatic speaker verification." in *Interspeech*, 2016, pp. 3628–3632.

[25] M. Todisco, H. Delgado, and N. Evans, "Constant q cepstral coefficients: A spoofing countermeasure for automatic speaker verification," *Computer Speech & Language*, vol. 45, pp. 516–535, 2017.

[26] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with sincnet," in *Proc. IEEE SLT*. IEEE, 2018, pp. 1021–1028.

[27] J.-w. Jung, H.-S. Heo, J.-h. Kim, H.-j. Shim, and H.-J. Yu, "Rawnet: Advanced end-to-end deep neural network using raw waveforms for text-independent speaker verification," *arXiv preprint arXiv:1904.08104*, 2019.

[28] S. Chen, Y. Wu, C. Wang, Z. Chen, Z. Chen, S. Liu, J. Wu, Y. Qian, F. Wei, J. Li *et al.*, "Unispeech-sat: Universal speech representation learning with speaker aware pre-training," *arXiv preprint arXiv:2110.05752*, 2021.

[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. NIPS*, 2017, pp. 5998–6008.

[30] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. CVPR*, 2018, pp. 7132–7141.

[31] S. Gao, M.-M. Cheng, K. Zhao, X.-Y. Zhang, M.-H. Yang, and P. H. Torr, "Res2net: A new multi-scale backbone architecture," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[32] L. Pepino, P. Riera, and L. Ferrer, "Emotion Recognition from Speech Using wav2vec 2.0 Embeddings," in *Proc. Interspeech 2021*, 2021, pp. 3400–3404.

[33] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proc. CVPR*, 2019, pp. 4690–4699.

[34] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[35] D. Snyder, G. Chen, and D. Povey, "MUSAN: A Music, Speech, and Noise Corpus," 2015, arXiv:1510.08484v1.

[36] J. Thienpondt, B. Desplanques, and K. Demuynck, "The idlab voxsrc-20 submission: Large margin fine-tuning and quality-aware score calibration in dnn based speaker verification," in *Proc. IEEE ICASSP 2021*. IEEE, 2021, pp. 5814–5818.

[37] Z. N. Karam, W. M. Campbell, and N. Dehak, "Towards reduced false-alarms using cohorts," in *Proc. IEEE ICASSP 2011*. IEEE, 2011, pp. 4512–4515.

[38] S. Cumani, P. D. Batzu, D. Colibro, C. Vair, P. Laface, and V. Vasilakakis, "Comparison of speaker recognition approaches for real applications." in *INTERSPEECH*, 2011, pp. 2365–2368.